– EPE – NEW NAME NEW DESIGN!

Practical Electronics

The UK's premier electronics and computing maker magazine

Audio Out
Wavecor crossover

Electronic Building Blocks
Loud voice alarm

Circuit Surgery
Strain gauge circuit revisited

Micromite
Adding Bluetooth functionality

Build this superb diode curve plotter

I/V Test
Reverse
Back
Zener Diode Tester

XOD ARDUINO

Exciting new series! Visual programming for Arduino with XOD

WIN! Microchip SAM D20 Xplained Pro Evaluation Kit

Bluetooth – create wireless projects for your Micromite

SILICON CHIP
STEAM WHISTLE

Toot toot! Steam whistle generator

Awesome Audio DSP

PLUS!

PIC n' Mix – Audio Spectrum Analyser design update
Net Work – Two-factor authentication and SSDs
Techno Talk – Boom time for battery traction

Mar 2020 £4.99
03
9 772632 573016

www.epemag.com | @practicalelec | practicalelectronics

With Over 250 Million Units in Stock, We've Got You Covered

Delivering You the World's Largest Inventory of Microchip Products

www.microchip.com/learn-more

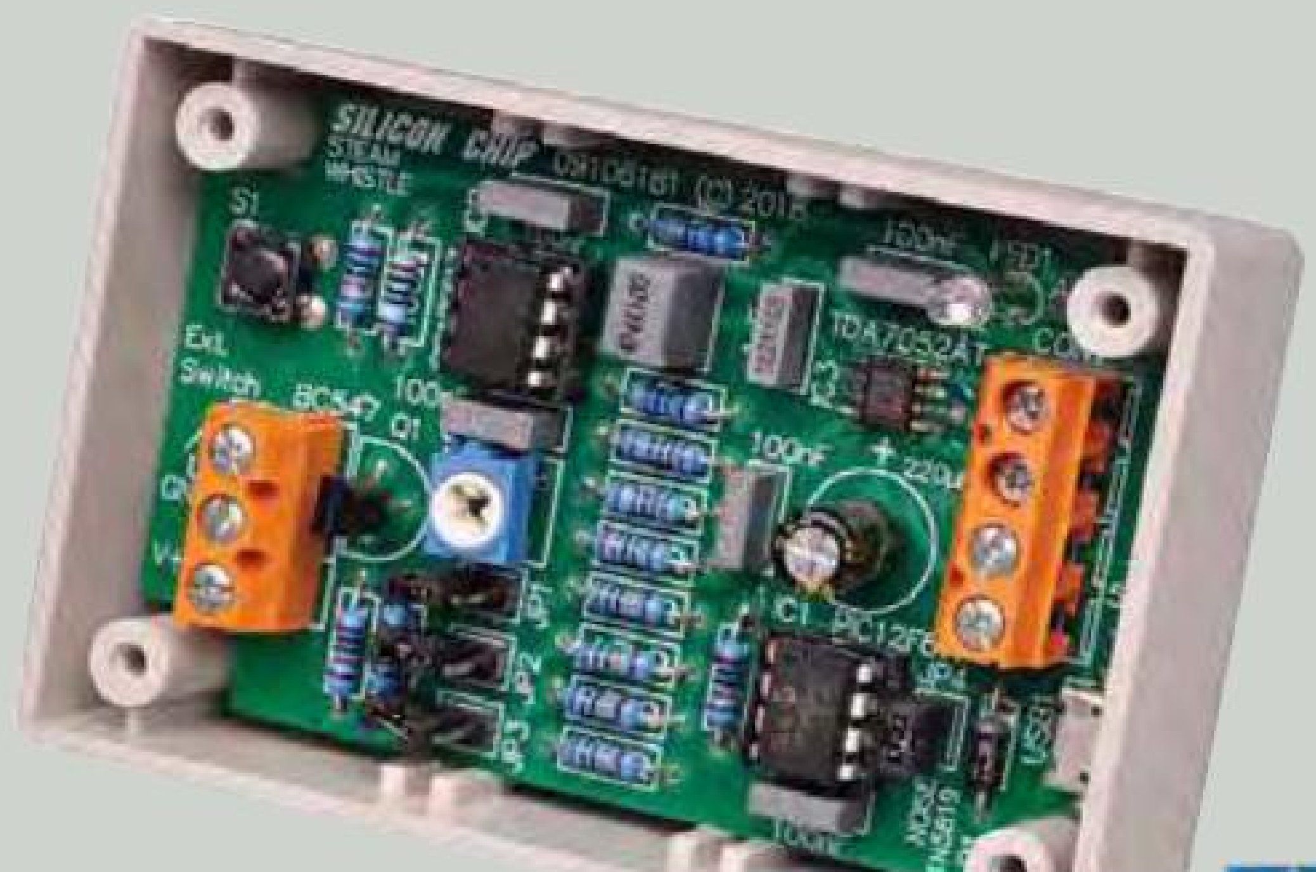
MICROCHIP

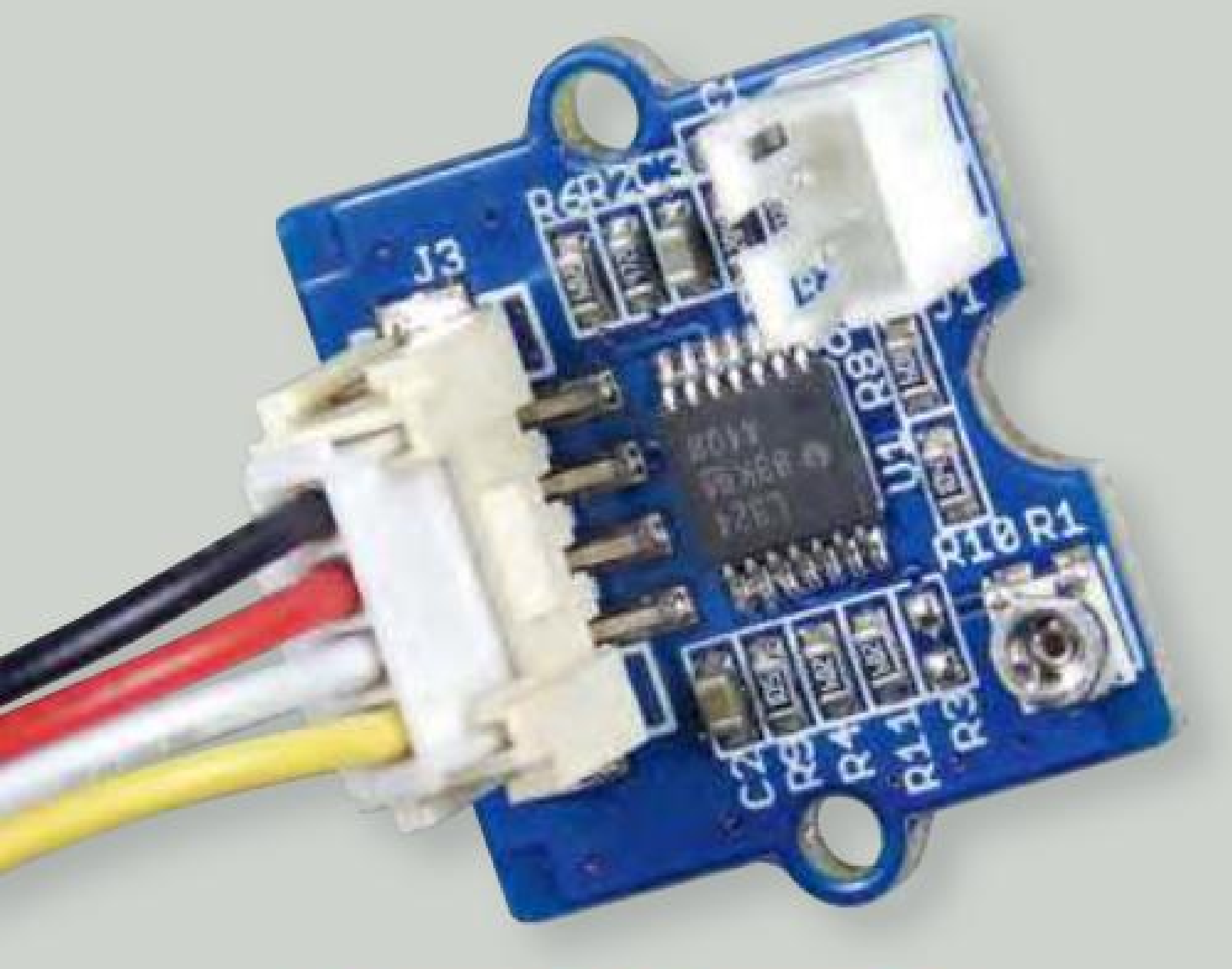
The Microchip name and logo and the Microchip logo are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.
All other trademarks are the property of their registered owners.
© 2018 Microchip Technology Inc. All rights reserved. DS00002552A. MEC2226Eng10/18

Practical
Electronics

Volume 49. No. 3
March 2020
ISSN 2632 573X

Contents





Made in the UK.

Written in Britain, Australia, the US and Ireland.

Read everywhere.

© Electron Publishing Limited 2020

Copyright in all drawings, photographs, articles, technical designs, software and intellectual property published in *Practical Electronics* is fully protected, and reproduction or imitation in whole or in part are expressly forbidden.

The April 2020 issue of *Practical Electronics* will be published on Thursday, 5 March 2020 – see page 80.

Projects and Circuits

Diode Curve Plotter by Tim Blythman 14
This superb piece of test gear is both practical and educational. It tests zener diodes, LEDs, schottky diodes, regular diodes, transient voltage suppressors and more.

Audio DSP – Part 3 by Phil Prosser (design) and Nicholas Vinen (words) 22
In this third and final article, we test the DSP Crossover modules, then connect them together and power the whole unit up so that it can process sound.

Steam Train Whistle / Diesel Horn Sound Generator by John Clarke 30
Relive the exciting days of steam-train travel with this sound generator, which can be customised in a number of ways to suit your project's preferences.

Using Cheap Asian Electronic Modules – Part 22 by Jim Rowe 36
Learn to use the Seeed/Grove-designed galvanic skin response sensor.

Series, Features and Columns

The Fox Report by Barry Fox 8
Easy when you know how, or have a decent user manual!

Techno Talk by Mark Nelson 10
Boom time for battery traction

Net Work by Alan Winstanley 12
The latest on Internet-based fraud, ID theft and advice on using 2FA (two-factor authentication) in your online dealings. Plus, SSDs – can you really entrust them?

PIC n' Mix by Mike Hibbett 40
Audio Spectrum Analyser – design update

Circuit Surgery by Ian Bell 44
Strain gauge circuit revisited

Audio Out by Jake Rothman 48
PE Mini-monitor crossover for Wavecor drivers – Part 2

Make it with Micromite by Phil Boyce 53
Part 14: Using Bluetooth to set your Micromite free

Visual programming with XOD by Julian Edgar 60
Introducing XOD visual programming for Arduino

Max's Cool Beans by Max The Magnificent 65
Flashing LEDs and drooling engineers

Electronic Building Blocks by Julian Edgar 76
High-powered voice alarm

Regulars and Services

Teach-In bundle – what a bargain! 3
Subscribe to Practical Electronics and save money 4
Wireless for the Warrior 6
Reader services – Editorial and Advertising Departments 7
Editorial 7
Not-so-smart speakers
Exclusive Microchip reader offer 11
Win a Microchip SAM D20 Xplained Pro Evaluation Kit
Practical Electronics back issues CD-ROM – great 15-year deal! 35
Practical Electronics – get your back issues here! 43
PE Teach-In 8 64
Practical Electronics CD-ROMS for electronics 70
A superb range of CD-ROMs for hobbyists, students and engineers
Direct Book Service 73
Build your library of carefully chosen technical books
Practical Electronics PCB Service 78
PCBs for Practical Electronics projects
Classified ads and Advertiser index 79
Next month! – highlights of our next issue of Practical Electronics 80

QUASAR
electronics

Quasar Electronics Limited
PO Box 6935, Bishops Stortford
CM23 4WP, United Kingdom

Tel: 01279 467799
E-mail: sales@quasarelectronics.co.uk
Web: quasarelectronics.co.uk

All prices INCLUDE 20.0% VAT. Free UK delivery on orders over £48
Postage & Packing Options (Up to 0.5Kg gross weight): **UK Standard 3-7 Day Delivery -** £4.95; **UK Mainland Next Day Delivery -** £9.95; **Europe (EU) -** £12.95; **Rest of World** - £14.95 (up to 0.5Kg).
!! Order online for reduced price postage and fast despatch !!
Payment: **We accept all major credit/debit cards. Make cheques/PO's payable to Quasar Electronics Limited.**
Please visit our online shop now for full details of over 1000 electronic kits, projects, modules and publications. Discounts for bulk quantities.

QuasarElectronics.co.uk

Solutions for Home, Education & Industry Since 1993

cebek

velleman

Official Main Dealer stocking the full range of Kits, Modules, Robots, Instruments, tools and much, much more...

PIC & ATMEL Programmers

We have a wide range of PIC, ATMEL Arduino and Raspberry Pi projects.

PIC Programmer & Experimenter Board

Great learning tool. Includes programming examples and a reprogrammable 16F627 Flash Microcontroller. Test buttons & LED indicators. Software to compile & program your source code is included. Supply: 12-15Vdc. Pre-assembled and ready to use.
*Order Code: VM111 - ~~**£38.88**~~ **£35.94***

USB PIC Programmer and Tutor Board

The only tutorial project board you need to take your first steps into Microchip PIC programming using a PIC16F882 (included). Later you can use it for more advanced programming. Programs all the devices a Microchip PICKIT2® can! Use the free Microchip tools for PICKit2™ & MPLAB® IDE environment.
*Order Code: EDU10 - **£46.74***

USB /Serial Port PIC Programmer

Fast programming. Wide range of PICs supported (see website for details). Free Windows software & ICSP header cable. USB or Serial connection. ZIF Socket, leads, PSU not included.
*Kit Order Code: 3149EKT - ~~**£49.96**~~ **£29.95***
*Assembled Order Code: AS3149E - **£44.95***
*Assembled with ZIF socket Order Code: AS3149EZIF - ~~**£74.96**~~ **£49.95***

PICKit™2 USB PIC Programmer Module

Versatile, low cost, PICKit™2 Development Programmer. Programs all the devices a Microchip PICKIT2 programmer can. Onboard sockets & ICSP header. USB powered.
*Assembled Order Code: VM203 - **£35.94***

Controllers & Loggers

Here are just a few of the controller and data acquisition and control units we have. See website for full details. 12Vdc PSU for all units: Order Code 660.446UK £10.68

USB Experiment Interface Board

Updated Version! 5 digital inputs, 8 digital outputs plus two analogue inputs and two analogue outputs. 8 bit resolution. DLL.
*Kit Order Code: K8055N - ~~**£39.95**~~ **£22.20***
*Assembled Order Code: VM110N - **£35.94***

2-Channel High Current UHF RC Set

State-of-the-art high security. Momentary or latching relay outputs rated to switch up to 240Vac @ 12 Amps. Range up to 40m. 15 Tx's can be learnt by one Rx. Kit includes one Tx (more available separately). 9-15Vdc.
*Kit Order Code: 8157KT - **£44.95***
*Assembled Order Code: AS8157 - **£49.96***

Computer Temperature Data Logger

Serial port 4-ch temperature logger. °C/°F. Continuously log up to 4 sensors located 200m+ from board. Choice of free software applications downloads for storing/using data. PCB just 45x45mm. Powered by PC. Includes one DS18S20 sensor.
*Kit Order Code: 3145KT - ~~**£19.95**~~ **£16.97***
*Assembled Order Code: AS3145 - **£19.96***
*Additional DS18S20 Sensors - **£4.96 each***

8-Channel Ethernet Relay Card Module

Connect to your router with standard network cable. Operate the 8 relays or check the status of input from anywhere in world. Use almost any internet browser, even mobile devices. Email status reports, programmable timers... Test software & DLL online.
*Assembled Order Code: VM201 - **£130.80***

Computer Controlled / Standalone Unipolar Stepper Motor Driver

Drives any 5-35Vdc 5, 6 or 8-lead unipolar stepper motor rated up to 6 Amps. Provides speed and direction control. Operates in stand-alone or PC-controlled mode for CNC use. Connect up to six boards to a single parallel port. Board supply: 9Vdc. PCB: 80x50mm.
*Kit Order Code: 3179KT - **£15.26***
*Assembled Order Code: AS3179 - **£22.26***

Card Sales & Enquiries 01279 467799

Bidirectional DC Motor Speed Controller

Control the speed of most common DC motors (rated up to 32Vdc/5A) in both the forward and reverse directions. The range of control is from fully OFF to fully ON in both directions. The direction and speed are controlled using a single potentiometer. Screw terminal block for connections. PCB: 90x42mm.
*Kit Order Code: 3166KT - **£19.99***
*Assembled Order Code: AS3166 - **£29.99***

8-Ch Serial Port Isolated I/O Relay Module

Computer controlled 8 channel relay board. 5A mains rated relay outputs and 4 opto-isolated digital inputs (for monitoring switch states, etc). Useful in a variety of control and sensing applications. Programmed via serial port (use our free Windows interface, terminal emulator or batch files). Serial cable can be up to 35m long. Includes plastic case 130x100x30mm. Power: 12Vdc, 500mA.
*Kit Order Code: 3108KT - **£74.95***
*Assembled Order Code: AS3108 - **£89.95***

8-Channel RF Remote Control Set

Control 8 onboard relays with included RF remote control unit. Toggle or momentary mode for each output. Up to 50m range. Board Supply: 12Vac, 500mA
*Assembled Order Code: VM118 - **£71.94***

Temperature Monitor & Relay Controller

Computer serial port temperature monitor & relay controller. Accepts up to four Dallas DS18S20 / DS18B20 digital thermometer sensors (1 included). Four relay outputs are independent of the sensors giving flexibility to setup the linkage any way you choose. Commands for reading temperature / controlling relays are simple text strings sent using a simple terminal or coms program (e.g. HyperTerminal) or our free Windows application. Supply: 12Vdc.
*Kit Order Code: 3190KT - ~~**£79.96**~~ **£47.95***
*Assembled Order Code: AS3190 - **£59.95***

3x5Amp RGB LED Controller with RS232

3 independent high power channels. Preprogrammed or user-editable light sequences. Standalone or 2-wire serial interface for microcontroller or PC communication with simple command set. Suits common anode RGB LED strips, LEDs, incandescent bulbs. 12A total max. Supply: 12Vdc. 69x56x18mm
*Kit Order Code: 8191KT - **£24.95***
*Assembled Order Code: AS8191 - **£27.95***

ETI BUNDLE (1) Teach-In 3, 4 and 5 – all on CD-ROM – only £18.95

ELECTRONICS TEACH-IN 3 – CD-ROM

Mike & Richard Tooley

FREE CD-ROM
ELECTRONICS TEACH-IN 3
£7.99
CIRCUIT SURGERY
PRACTICALLY SPEAKING
INGENUITY UNLIMITED
FREE TEACH-IN 1 CD-ROM
TWO TEACH-INs FOR THE PRICE OF ONE!

The three sections of the *Teach-In 3* CD-ROM cover a huge range of subjects that will interest everyone involved in electronics – from newcomers to the hobby and students to experienced constructors and professionals.

The first section (80 pages) is dedicated to *Circuit Surgery*, *EPE/PE's* regular clinic dealing with readers' queries on circuit design problems – from voltage regulation to using SPICE circuit simulation software.

The second section – *Practically Speaking* – covers hands-on aspects of electronics construction. Again, a whole range of subjects, from soldering to avoiding problems with static electricity and identifying components is covered. Finally, our collection of *Ingenuity Unlimited* circuits provides over 40 circuit designs submitted by readers.

The CD-ROM also contains the complete *Electronics Teach-In 1* book, which provides a broad-based introduction to electronics in PDF form, plus interactive quizzes to test your knowledge and TINA circuit simulation software (a limited version – plus a specially written TINA Tutorial).

The *Teach-In 1* series covers everything from electric current through to microprocessors and microcontrollers, and each part includes demonstration circuits to build on breadboards or to simulate on your PC.

ELECTRONICS TEACH-IN 4 – CD-ROM A BROAD-BASED INTRODUCTION TO ELECTRONICS

Mike & Richard Tooley

FREE CD-ROM
ELECTRONICS TEACH-IN 4
£8.99
A BROAD-BASED INTRODUCTION TO ELECTRONICS
THE MODERN ELECTRONICS MANUAL
BASE MANUAL

The *Teach-In 4* CD-ROM covers three of the most important electronics units that are currently studied in many schools and colleges. These include, Edexcel BTEC level 2 awards and the electronics units of the Diploma in Engineering, Level 2.

The CD-ROM also contains the full *Modern Electronics Manual*, worth £29.95. **The Manual** contains over 800 pages of electronics theory, projects, data, assembly instructions and web links.

A package of exceptional value that will appeal to anyone interested in learning about electronics – hobbyists, students or professionals.

ELECTRONICS TEACH-IN 5 – CD-ROM JUMP START

Mike & Richard Tooley

FREE CD-ROM
ELECTRONICS TEACH-IN 5
£8.99
JUMP START
PRACTICALLY SPEAKING
PIC N' MIX
TEACH-IN 2
TWO TEACH-INs FOR THE PRICE OF ONE

15 design and build circuit projects for newcomers or those following courses in school and colleges.

The projects are: ■ Moisture Detector ■ Quiz Machine ■ Battery Voltage Checker ■ Solar-Powered Charger ■ Versatile Theft Alarm ■ Spooky Circuits ■ Frost Alarm ■ Mini Christmas Lights ■ iPod Speaker ■ Logic Probe ■ DC Motor Controller ■ Egg Timer ■ Signal Injector Probe ■ Simple Radio Receiver ■ Temperature Alarm.

PLUS

PIC'n' Mix – starting out with the popular range of PIC microcontrollers and *Practically Speaking* – tips and techniques for project construction.

The CD-ROM also contains:

- ■ Complete *Teach-In 2* book, a practical introduction to PIC microprocessors
- ■ MikroElektronika, Microchip and L-Tek PoScope software.

Three Teach-ins for the great price of £18.95

PLUS you also get the contents of the free CD-ROM from each issue...

... so that's another TWO Teach-Ins and The Full Modern Electronics Manual!

What a Bargain!!

ETI BUNDLE (2) Teach-In 6, 7 and 8 – all on CD-ROM – only £18.95

ELECTRONICS TEACH-IN 6 – CD-ROM A COMPREHENSIVE GUIDE TO RASPBERRY Pi

Mike & Richard Tooley

Teach-In 6 contains an exciting series of articles that provides a complete introduction to the Raspberry Pi, the low cost computer that has taken the education and computing world by storm.

This latest book in our *Teach-In* series will appeal to electronics enthusiasts and computer buffs who want to get to grips with the Raspberry Pi.

Teach-In 6 is for anyone searching for ideas to use their Pi, or who has an idea for a project but doesn't know how to turn it into reality. This book will prove invaluable for anyone fascinated by the revolutionary Pi. It covers:

- ■ Pi programming
- ■ Pi hardware
- ■ Pi communications
- ■ Pi Projects
- ■ Pi Class
- ■ Python Quickstart
- ■ Pi World
- ■ ...and much more!

The *Teach-In 6* CD-ROM also contains all the necessary software for the series, so that readers and circuit designers can get started quickly and easily with the projects and ideas covered.

FREE DVD-ROM
ELECTRONICS TEACH-IN 6
£8.99
RASPBERRY Pi
A COMPREHENSIVE GUIDE TO RASPBERRY Pi
PLUS

ELECTRONICS TEACH-IN 7 – CD-ROM DISCRETE LINEAR CIRCUIT DESIGN

Mike & Richard Tooley

Teach-In 7 is a complete introduction to the design of analogue electronic circuits. It is ideal for everyone interested in electronics as a hobby and for those studying technology at schools and colleges. The CD-ROM also contains all the circuit software for the course, plus demo CAD software for use with the *Teach-In* series.

- ■ Discrete Linear Circuit Design
- ■ Understand linear circuit design
- ■ Learn with 'TINA' – modern CAD software
- ■ Design simple, but elegant circuits
- ■ Five projects to build:
  - i) Pre-amp
  - ii) Headphone Amp
  - iii) Tone Control
  - iv) VU-meter
  - v) High Performance Audio Power Amp.

PLUS

Audio Out – an analogue expert's take on specialist circuits

Practically Speaking – the techniques of project building.

FREE CD-ROM
ELECTRONICS TEACH-IN 7
£8.99
DISCRETE LINEAR CIRCUIT DESIGN
PLUS...
AUDIO OUT
PRACTICALLY SPEAKING

ELECTRONICS TEACH-IN 8 – CD-ROM INTRODUCING THE ARDUINO

Mike & Richard Tooley

Hardware: learn about components and circuits
Programming: powerful integrated development system
Microcontrollers: understand control operations
Communications: connect to PCs and other Arduinos.

Teach-In 8 is an exciting series designed for electronics enthusiasts who want to get to grips with the inexpensive, popular Arduino microcontroller, as well as coding enthusiasts who want to explore hardware and interfacing. It will provide a one-stop source of ideas and practical information.

The Arduino offers a truly effective platform for developing a huge variety of projects; from operating a set of Christmas tree lights to remotely controlling a robotic vehicle through wireless or the Internet.

Teach-In 8 is based around a series of practical projects with plenty of information for customisation.

This book also includes *PIC n' Mix*: 'PICs and the PICkit 3 – A Beginners guide' by Mike O'Keefe and *Circuit Surgery* by Ian Bell – 'State Machines part 1 and 2'.

The CD-ROM includes the files for:

- ■ *Teach-In 8*
- ■ Microchip MPLAB IDE XC8 8-bit compiler
- ■ PICkit 3 User Guide
- ■ Lab-Nation Smartscope software.

FREE CD-ROM
ELECTRONICS TEACH-IN 8
£8.99
INTRODUCING THE ARDUINO
PLUS...

ORDER YOUR BUNDLE TODAY!
JUST CALL 01202 880299 – OR VISIT www.epemag.com

– EPE – NEW NAME NEW DESIGN!

Practical Electronics

UK readers SAVE £1 on every issue

NEW subscriptions hotline!

01202 087631

Take out a one-year subscription and save £10 over the year.
Even better – save £1 per issue if you subscribe for two years – a total saving of £24.

Overseas rates represent exceptional value

You also:
- **Avoid any cover price increase for the duration of your subscription**
- **Get your magazine delivered to your door each month**
- **Guarantee a copy, even if the newsagents sell out**

Order by:
- **Phone or post with a cheque, postal order or credit card**
- **Or order online at: www.epemag.com (go to our secure server-backed online shop)**

Practical Electronics

SUBSCRIPTION PRICES

Subscriptions for delivery direct to any address in:

UK: 6-months £26.99, 12-months £49.85, 24-months £94.99

Europe Airmail: 6-months £30.99, 12-months £57.99, 24-months £109.99

Rest Of The World Airmail: 6-months £37.99, 12-months £70.99, 24-months £135.99

Cheques (in **£ sterling only**) payable to *Practical Electronics* and sent to: Practical Electronics Subscriptions, PO Box 6337, Bournemouth BH1 9EH, United Kingdom

Tel: 01202 087631
Email: pesubs@selectps.com

Also via our secure online shop at**: www.epemag.com**

Subscriptions start with the next available issue. We accept MasterCard or Visa.
(For past issues see the Back Issues page.)

DIGITAL SUBSCRIPTIONS

Electronic subscriptions (PDF download or online) are available from £20.99 for 12 months, more details at: **www.epemag.com**

SUBSCRIPTION ORDER FORM

☐ 6 Months: UK £26.99, Europe £30.99 (Airmail), Rest of the World £37.99 (Airmail)
☐ 1 Year: UK £49.85, Europe £57.99 (Airmail), Rest of the World £70.99 (Airmail)
☐ 2 Years: UK £94.99, Europe £109.99 (Airmail), Rest of the World £135.99 (Airmail)

To: **Practical Electronics Subscriptions**
PO Box 6337
Bournemouth BH1 9EH
United Kingdom

Tel: 01202 087631
Email: pesubs@selectps.com

I enclose payment of £ ....
(cheque/PO in £ sterling only)
payable to *Practical Electronics*

☐ Please charge my Visa/Mastercard

My card number is: ....
Please print clearly, and check that you have the number correct

Card Security Code .... Valid From Date.......................
(The last 3 digits on or just under the signature strip)

Card Ex. Date ....

Name ....

Address ....

Post code .... Tel. ....

Email ....

Subscriptions start with the next available issue.

Process Control
Automatic mode

Digitally Enhance Your Analog Controls

Combine the Speed of an Analog Controller With the Flexibility of a Digital Microcontroller

CHARGER
CHARGE
END
LITHIUM ION
BATTERY PACK
7.4V 1000mAh

No system can perform without reliable power supplies. Our Digitally Enhanced Power Analog (DEPA) family of products combines the performance of an analog Pulse-Width Modulation (PWM) controller with the configurability of an 8-bit PIC® microcontroller (MCU). The combination of these methods allows the addition of digital features to a reliable, easy-to-implement analog control loop, including fast transient responses, high efficiencies, reliable gain and phase margins. Adding the ability to measure and respond to changes with tailored algorithms improves the robustness of the system, while offering diagnostic and communication options.

The single-chip solution can accept a high-voltage input and regulate a wide output current or voltage range, which allows you to maintain robust operation within an unstable environment. Discover how the flexibility of our DEPA products can enhance your next design.

Key Features

- Fast and efficient power conversion with analog current-mode control loop
- Flexible control with an integrated MCU
- Dynamically settable hardware protections enable robust operation

MICROCHIP
MCP19117

www.microchip.com/FlexiblePower

MICROCHIP

The Microchip name and logo and the Microchip logo are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries. All other trademarks are the property of their registered owners.
© 2019 Microchip Technology Inc. All rights reserved. DS20006124A. MEC2309A-ENG-12-19

WIRELESS FOR THE WARRIOR

by LOUIS MEULSTEE

THE DEFINITIVE TECHNICAL HISTORY OF RADIO COMMUNICATION EQUIPMENT IN THE BRITISH ARMY

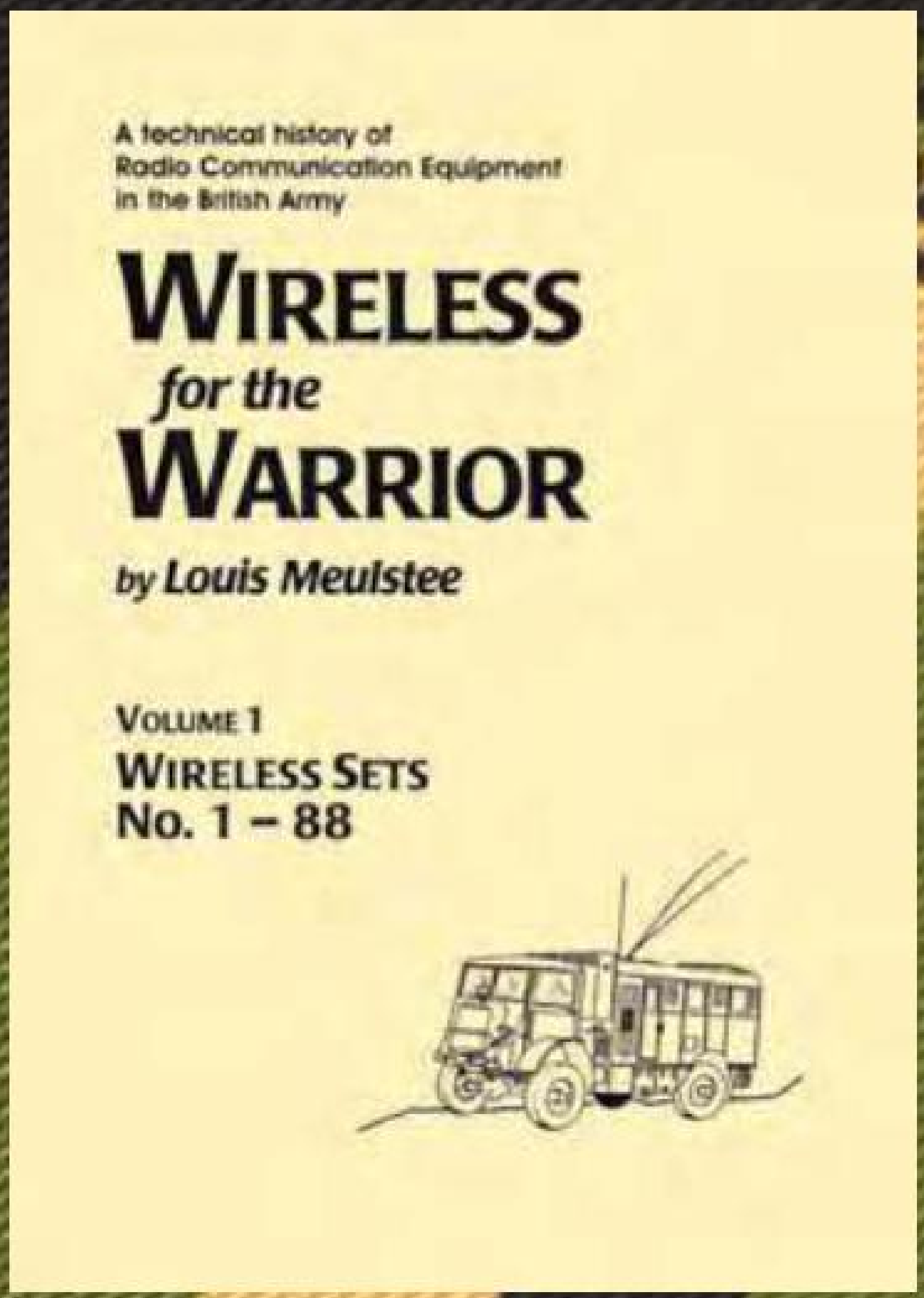
The *Wireless for the Warrior* books are a source of reference for the history and development of radio communication equipment used by the British Army from the very early days of wireless up to the 1960s.

The books are very detailed and include circuit diagrams, technical specifications and alignment data, technical development history, complete station lists and vehicle fitting instructions.

Volume 1 and *Volume 2* cover transmitters and transceivers used between 1932-1948. An era that starts with positive steps taken to formulate and develop a new series of wireless sets that offered great improvements over obsolete World War I pattern equipment. The other end of this timeframe saw the introduction of VHF FM and hermetically sealed equipment.

Volume 3 covers army receivers from 1932 to the late 1960s. The book not only describes receivers specifically designed for the British Army, but also the Royal Navy and RAF. Also covered: special receivers, direction finding receivers, Canadian and Australian Army receivers, commercial receivers adopted by the Army, and Army Welfare broadcast receivers.

Volume 4 covers clandestine, agent or 'spy' radio equipment, sets which were used by special forces, partisans, resistance, 'stay behind' organisations, Australian Coast Watchers and the diplomatic service. Plus, selected associated power sources, RDF and intercept receivers, bugs and radar beacons.



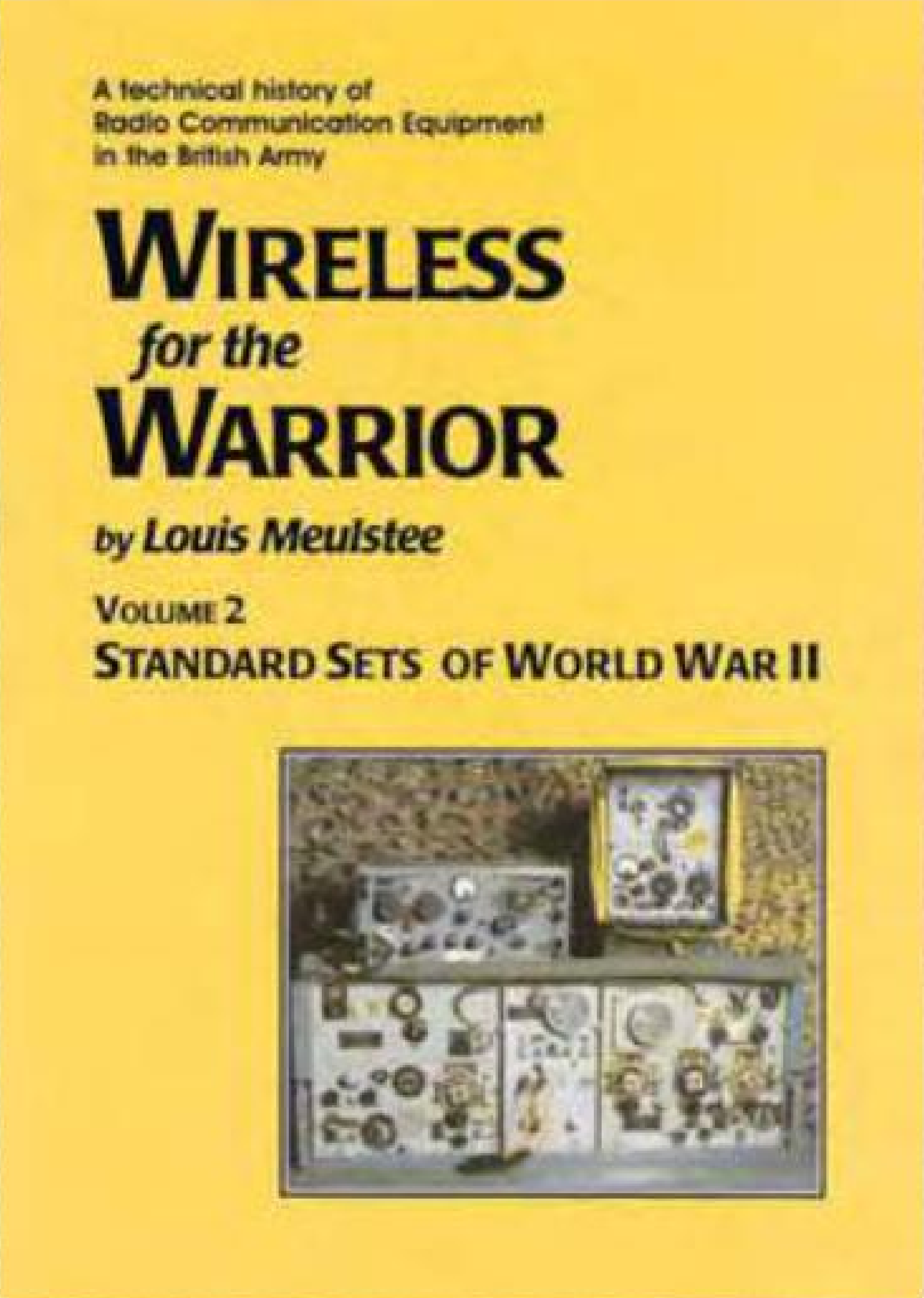
A technical history of Radio Communication Equipment in the British Army

WIRELESS for the WARRIOR

by Louis Meulstee

VOLUME 1

WIRELESS SETS No. 1 – 88



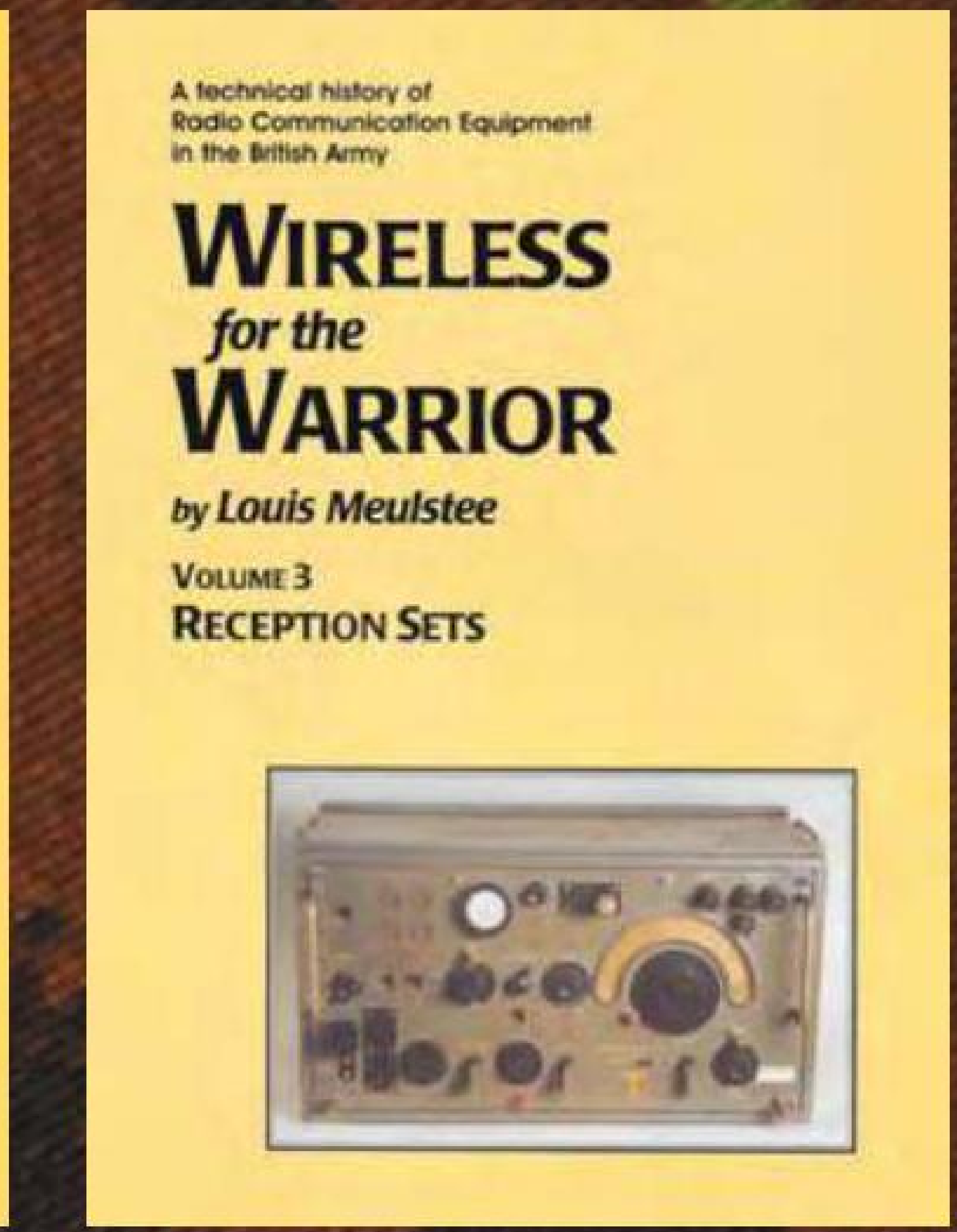
A technical history of Radio Communication Equipment in the British Army

WIRELESS for the WARRIOR

by Louis Meulstee

VOLUME 2

STANDARD SETS OF WORLD WAR II



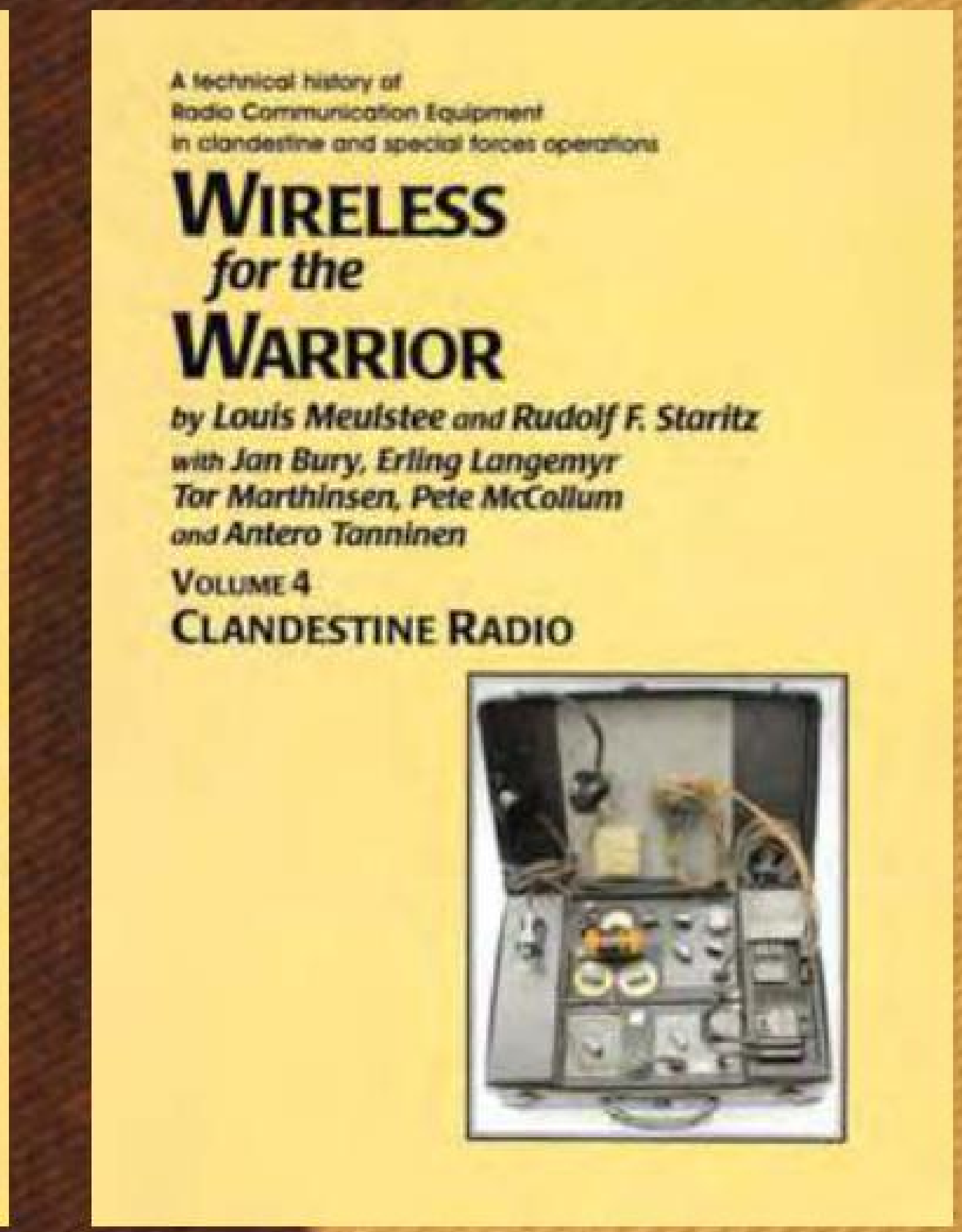
A technical history of Radio Communication Equipment in the British Army

WIRELESS for the WARRIOR

by Louis Meulstee

VOLUME 3

RECEPTION SETS



A technical history of Radio Communication Equipment in clandestine and special forces operations

WIRELESS for the WARRIOR

by Louis Meulstee and Rudolf F. Staritz with Jan Bury, Erling Langemyr Tor Marthinsen, Pete McCollum and Antero Tanninen

VOLUME 4

CLANDESTINE RADIO

ORDER YOURS TODAY!
JUST CALL 01202 880299 OR VISIT www.epemag.com

Practical Electronics

Volume 49. No. 3
March 2020
ISSN 2632 573X

Editorial

Editorial offices

Practical Electronics
Electron Publishing Limited
1 Buckingham Road
Brighton
East Sussex BN1 3RA

Tel 01273 777619
Mob 07973 518682
Fax 01202 843233
Email pe@electronpublishing.com
Web www.epemag.com

Advertisement offices

Practical Electronics Adverts
1 Buckingham Road
Brighton
East Sussex BN1 3RA

Tel 01273 777619
Mob 07973 518682
Email pe@electronpublishing.com

Editor	Matt Pulzer		
General Manager	Louisa Pulzer		
Digital subscriptions	Stewart Kearn	**Tel**	01202 880299
Online Editor	Alan Winstanley		
Web Systems	Kris Thain		
Publisher	Matt Pulzer		

Print subscriptions

Practical Electronics Subscriptions
PO Box 6337
Bournemouth BH1 9EH
United Kingdom

Tel 01202 087631
Email pesubs@selectps.com

Technical enquiries

We regret technical enquiries cannot be answered over the telephone. We are unable to offer any advice on the use, purchase, repair or modification of commercial equipment or the incorporation or modification of designs published in the magazine. We cannot provide data or answer queries on articles or projects that are more than five years old.

Questions about articles or projects should be sent to the editor by email: pe@electronpublishing.com

Projects and circuits

All reasonable precautions are taken to ensure that the advice and data given to readers is reliable. We cannot, however, guarantee it and we cannot accept legal responsibility for it.

A number of projects and circuits published in *Practical Electronics* employ voltages that can be lethal. You should not build, test, modify or renovate any item of mains-powered equipment unless you fully understand the safety aspects involved and you use an RCD (GFCI) adaptor.

Component supplies

We do not supply electronic components or kits for building the projects featured, these can be supplied by advertisers. We advise readers to check that all parts are still available before commencing any project in a back-dated issue.

Advertisements

Although the proprietors and staff of *Practical Electronics* take reasonable precautions to protect the interests of readers by ensuring as far as practicable that advertisements are bona fide, the magazine and its publishers cannot give any undertakings in respect of statements or claims made by advertisers, whether these advertisements are printed as part of the magazine, or in inserts. The Publishers regret that under no circumstances will the magazine accept liability for non-receipt of goods ordered, or for late delivery, or for faults in manufacture.

Transmitters/bugs/telephone equipment

We advise readers that certain items of radio transmitting and telephone equipment which may be advertised in our pages cannot be legally used in the UK. Readers should check the law before buying any transmitting or telephone equipment, as a fine, confiscation of equipment and/or imprisonment can result from illegal use or ownership. The laws vary from country to country; readers should check local laws.

Not-so-smart speakers

One of the big tech news stories the week I'm writing this Editorial is the howls of outrage that greeted the announcement by smart speaker manufacturer Sonos that they were no longer supporting some of their older products – and in some cases not-so-old – with software updates. Customers who had spent small fortunes on whole-house installations were furious that their investment was compromised and that at best they would need to run two incompatible systems to pipe music though their home. This came just eight months after Google announced it was to cease supporting its Works programme, which allowed third-party vendors to integrate products such as smart light bulbs with its Nest ecosystem of home automation. There are (at least) five conclusions one could draw from these examples.

First, no matter how big or well established the company, don't assume it's got your back for product support. Companies exist to make money, fight off rivals and survive in a fast-moving market. Being innovative or a market leader does not guarantee a future – just ask Nokia or Motorola. This makes them ruthless and paranoid. If it is no longer in a company's interest to write software for old products then they will simply stop. You may have a perfectly good scanner, but if your new PC OS upgrade doesn't have a driver for it then you're out of luck – that problem has bitten me twice.

Second, companies can be arrogant and stupid. I have no idea whether the decision by Sonos was technologically pragmatic and justified, or a truly cynical attempt to 'force' their loyal customer base to upgrade to the latest expensive hardware by deliberately rendering their existing products obsolete. I'm not inclined to give them the benefit of the doubt since one of the products involved is not even five years old. Either way, it seems like a spectacular own goal. Would you invest hundreds, even thousands with a company whose products might not work properly after a few years? No, me neither.

Third, as always, it's a case of buyer beware. Remember, if you are buying something with software then there will always be questions around updates and maintenance. If you are buying something with software *and* it needs to talk to other systems then you should really ask yourself will every software subsystem from communication via Bluetooth or Wi-Fi to its rudimentary 'operating system' continue to be maintained, patched and supported in the years ahead, or at least as long as the hardware works. In truth, this is pretty much a rhetorical question because the answer is almost always 'no'. All software has an expiration date, and all too often it will expire before the hardware gives up the ghost.

Fourth, 'all-in-one' systems like smart speakers are especially vulnerable to these problems, because it means that you could have a high-quality piece of expensive Hi-Fi that is effectively rendered useless because just one part no longer works.

Last, my music system runs off a PC via a DAC and then into a 20-year-old Naim pre-amp/power amp combo. It doesn't understand voice commands, but the audio parts will carry on working for another 20 years. How many smart speakers could make the same claim?

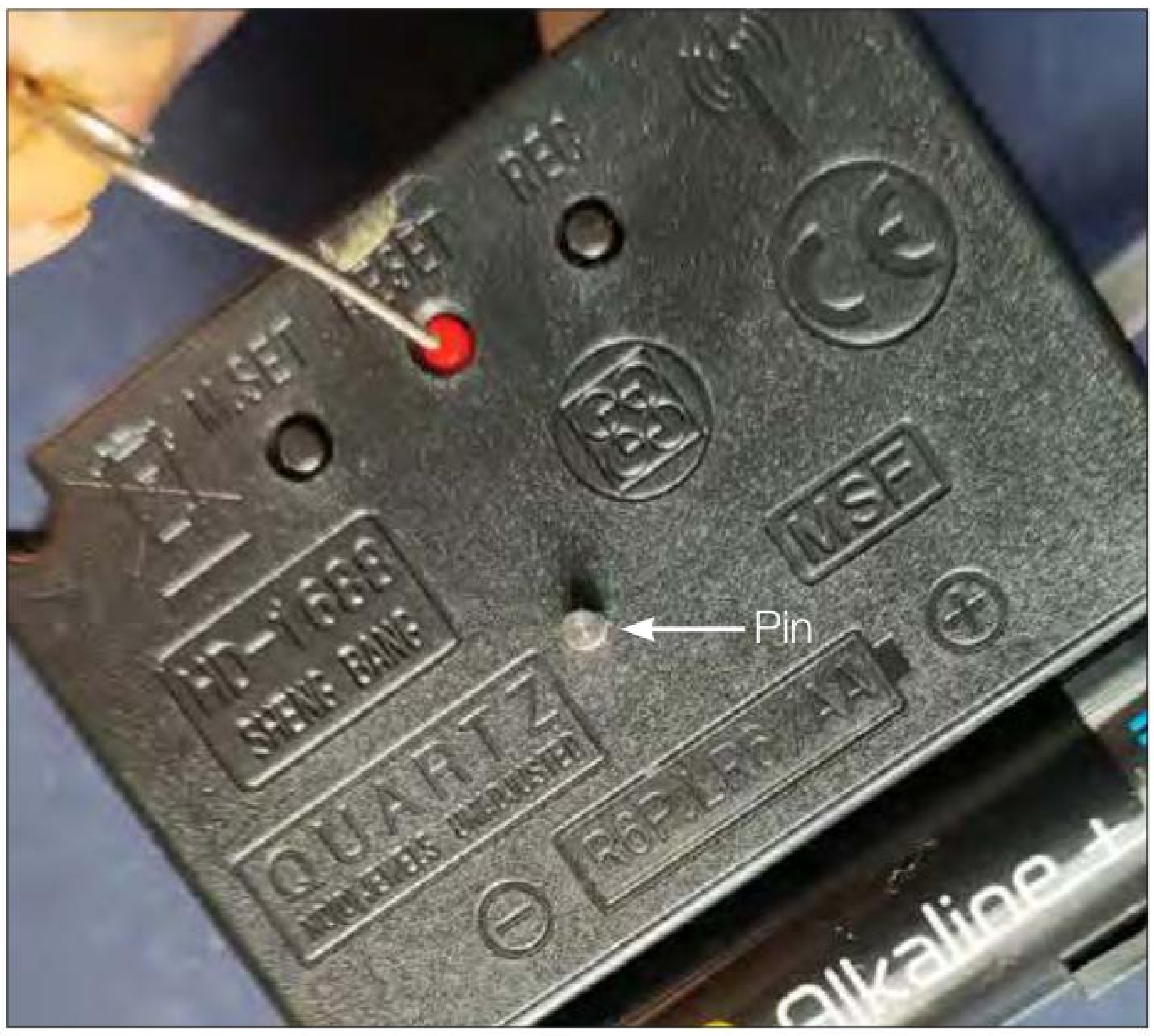
Matt Pulzer
Publisher

The **Fox** *Report*

Barry Fox's technology column

Easy when you know how, or have a decent user manual!

Instruction manuals got a bad name when early home computers were sold with thick volumes of gibberish that only made sense when you already knew what the instructions were trying to say. The pendulum has now swung to the other extreme; manuals so simplistic that they are largely useless, accompanied by a thick bundle of even more useless legal health and safety warnings. Two recent examples exemplify the pitfalls.



Shown here, one of many attempts to reset the clock. All failed until I removed the locking pin – the little silver object next to the 'quartz' label. I found it by chance when I opened the body to look for a non-existent fault, and the pin fell out.

Wasting time

I recently embarked on a project to replace the movements in several wall clocks with radio-controlled equivalents – and so ensure that clocks around the house were not an hour out for half the year. All the movements came with no instructions and none of them moved the hands when fitted with new 1.5V cells. The vendor, based in Britain, was obviously working from all-purpose scripted fob-off replies like, 'take them to the nearest repair shop'. I guess he has by now had to refund a lot of customers.

What I discovered by trial and error could have been simply written: 'There is a small, un-marked and inconspicuous locking pin in the casing which must be removed before a battery is inserted. Then wait for the bare spindles to turn quite fast and stop at a zero point. Next, before the spindles start to turn again, very carefully push-fit the hour, minute and second hands, all pointing to 12, without moving the spindles. Then leave the movement untouched to self-adjust to the NPL (National Physical Laboratory) 60kHz atomic clock signal broadcast from Anthorn in Cumbria. Note that to receive an adequate radio signal you may have to try several places in the room, near windows. And maybe leave it overnight. Do not try to move the hands after they have been fitted to the spindles. If you make a mistake and inadvertently move the hands, do it all over again.'

Not much to ask for is it? And I offer the above (without guarantee!) to help any readers who are contemplating a similar exercise.

Sounds easy, but…

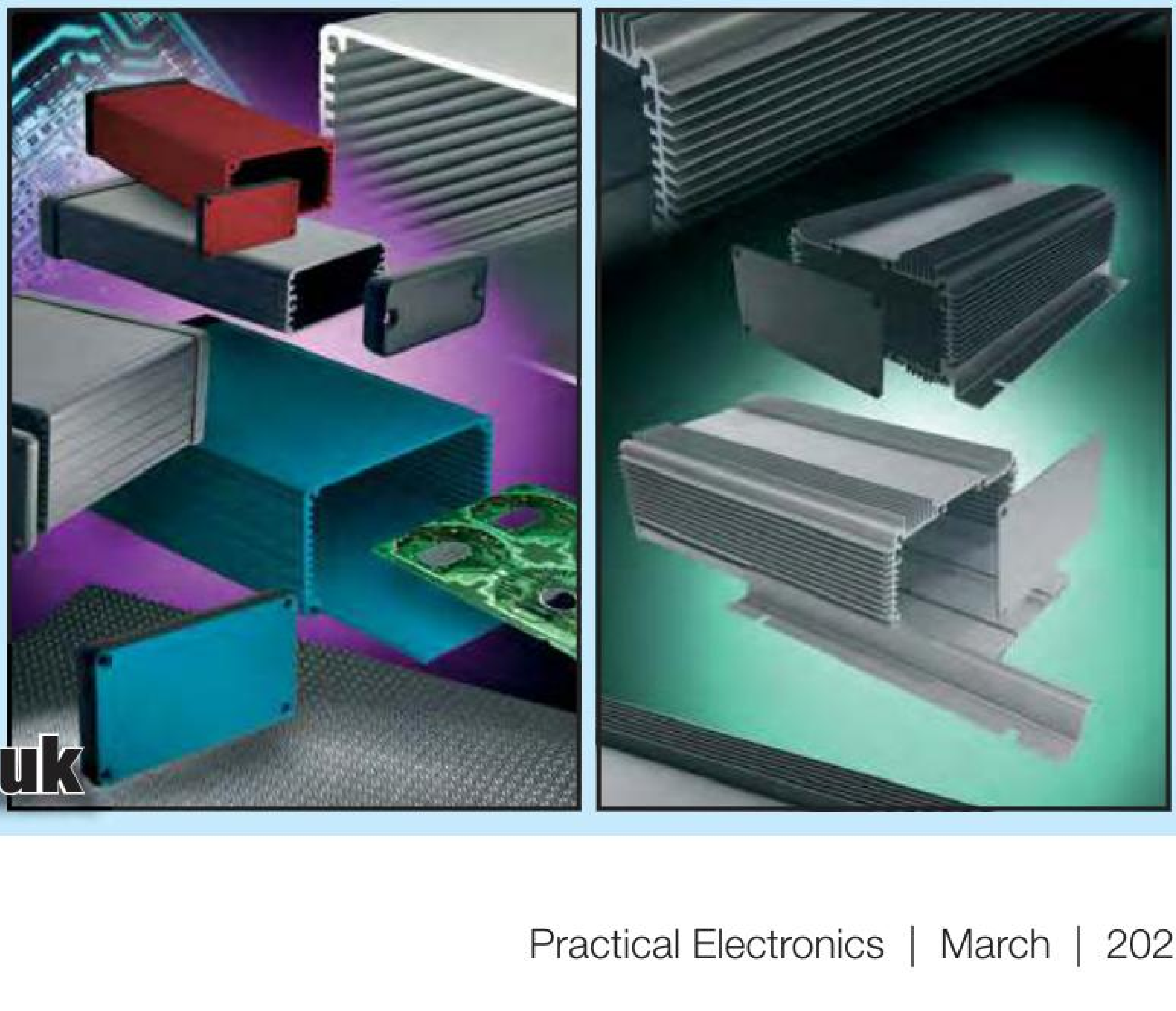
The native sound from flat-screen TVs and small PCs is usually very poor, so sales of sound bars and docks are booming. Unfortunately, the instructions are often well-nigh useless. They fail to tell users clearly what the devices can do, and how to get them to do it. And many users, who are accustomed to just watching TV, need to be told.

HAMMOND
MANUFACTURING®

Extruded enclosures
standard and heatsink

www.hammondmfg.com/1455.htm
www.hammondmfg.com/1455NHD.htm

01256 812812
sales@hammond-electronics.co.uk



I recently tested an A70 sound bar and E30 audio dock made by British company Orbitsound. The designs are British, the prices reasonable and neither follows the unappealing trend of trying to trick up the sound with pseudo surround and Atmos height. After time-consuming correspondence with Orbitsound's help line (my time and theirs) the company now acknowledges the need for better user manuals. Here are a few essential basics that really should be clearly set out in all user guides by all manufacturers.

The dock/bar connects either with a 3.5mm Aux analogue jack cable or digital optical SP/DIF cable to a TV or PC. Usually plugging the Aux jack into a TV/PC will mute its own speakers. Controlling volume and muting on the TV/PC will then control volume and muting from the dock/bar. But, to avoid hiss, hum and digital ticking noise, set the dock/bar volume at whatever level lets the TV/PC send out an average meaty level of around 80%.

Connecting by digital optical cable should give better sound, but the volume level coming from the dock/bar is fixed so sound volume must be controlled by the dock/bar and its own remote.

Teach your system

This is why a 'learn' option (as offered by Orbitsound and bars made by German company Canton) is useful. The bar/dock can be 'taught' to respond to volume and mute control commands from the TV's own remote. The TV speakers can be turned off and the TV remote controls the dock/bar volume.

Sometimes – as happened to me with the Orbitsound bar – the dock/bar may stubbornly refuse to learn commands because the TV remote uses too-complex coding. Fortunately there is a workaround that sometimes works. Use the troublesome remote to teach another remote with learning facility, and then use that remote to teach the bar/dock.

The bar/dock may also be able to connect with a TV/PC by Bluetooth. But the Bluetooth signal processing will add some delay to the audio. This 'latency' may be enough to destroy lip sync with a TV screen. The proprietary aptX coding system – if supported – will generally have lower latency than the basic Bluetooth standard SBC A2DP (Sub Band Codec, Advanced Audio Distribution Profile).

Barry Fox, FBKS (Fellow, International Moving Image Society)

In Search of Simulacra:
Modeling a Self-Learning Android

Photo: interloveupted.blogspo.com
Simulacra: an automated robot mentioned in Homer's Illiad – 700 to 800 BC

The reader is taken through basic understanding of human nature, thinking, learning, problem solving. Then Conceptual information about basic control systems through to Artificial Neural Networks and software architectures is presented. All in plain language. The book goes on to explain the details of how a self-learning Android could work by putting together those previously described control systems.

Available on Amazon.UK
Written in plain language, for anyone interested in the next step in Artificial Intelligence
www.amazon.co.uk/dp/1513653075

STEWART OF READING
17A King Street, Mortimer, near Reading, RG7 3RS
Telephone: **0118 933 1111** Fax: **0118 933 2375**
USED ELECTRONIC TEST EQUIPMENT
Check website **www.stewart-of-reading.co.uk**
(ALL PRICES PLUS CARRIAGE & VAT)
Please check availability before ordering or calling in

Fluke/Philips PM3092 Oscilloscope
2+2 Channel 200MHz Delay TB,
Autoset etc – £250

HP 54600B Oscilloscope
Analogue/Digital Dual Trace 100MHz
Only £75, with accessories £125

LAMBDA GENESYS	PSU GEN100-15 100V 15A Boxed As New	£400
LAMBDA GENESYS	PSU GEN50-30 50V 30A	£400
IFR 2025	Signal Generator 9kHz – 2.51GHz Opt 04/11	£900
IFR 2948B	Communication Service Monitor Opts 03/25 Avionics	POA
IFR 6843	Microwave Systems Analyser 10MHz – 20GHz	POA
R&S APN62	Syn Function Generator 1Hz – 260kHz	£295
Agilent 8712ET	RF Network Analyser 300kHz – 1300MHz	POA
HP8903A/B	Audio Analyser	£750 – £950
HP8757D	Scaler Network Analyser	POA
HP3325A	Synthesised Function Generator	£195
HP3561A	Dynamic Signal Analyser	£650
HP6032A	PSU 0-60V 0-50A 1000W	£750
HP6622A	PSU 0-20V 4A Twice or 0-50V 2A Twice	£350
HP6624A	PSU 4 Outputs	£400
HP6632B	PSU 0-20V 0-5A	£195
HP6644A	PSU 0-60V 3.5A	£400
HP6654A	PSU 0-60V 0-9A	£500
HP8341A	Synthesised Sweep Generator 10MHz – 20GHz	£2,000
HP83630A	Synthesised Sweeper 10MHz – 26.5 GHz	POA
HP83624A	Synthesised Sweeper 2 – 20GHz	POA
HP8484A	Power Sensor 0.01-18GHz 3nW-10μW	£75
HP8560E	Spectrum Analyser Synthesised 30Hz – 2.9GHz	£1,750
HP8563A	Spectrum Analyser Synthesised 9kHz – 22GHz	£2,250
HP8566B	Spectrum Analsyer 100Hz – 22GHz	£1,200
HP8662A	RF Generator 10kHz – 1280MHz	£750
Marconi 2022E	Synthesised AM/FM Signal Generator 10kHz – 1.01GHz	£325
Marconi 2024	Synthesised Signal Generator 9kHz – 2.4GHz	£800
Marconi 2030	Synthesised Signal Generator 10kHz – 1.35GHz	£750
Marconi 2023A	Signal Generator 9kHz – 1.2GHz	£700

Marconi 2305	Modulation Meter	£250
Marconi 2440	Counter 20GHz	£295
Marconi 2945/A/B	Communications Test Set Various Options	POA
Marconi 2955	Radio Communications Test Set	£595
Marconi 2955A	Radio Communications Test Set	£725
Marconi 2955B	Radio Communications Test Set	£800
Marconi 6200	Microwave Test Set	£1,500
Marconi 6200A	Microwave Test Set 10MHz – 20GHz	£1,950
Marconi 6200B	Microwave Test Set	£2,300
Marconi 6960B	Power Meter with 6910 sensor	£295
Tektronix TDS3052B	Oscilloscope 500MHz 2.5GS/s	£1,250
Tektronix TDS3032	Oscilloscope 300MHz 2.5GS/s	£995
Tektronix TDS3012	Oscilloscope 2 Channel 100MHz 1.25GS/s	£450
Tektronix 2430A	Oscilloscope Dual Trace 150MHz 100MS/s	£350
Tektronix 2465B	Oscilloscope 4 Channel 400MHz	£600
Farnell AP60/50	PSU 0-60V 0-50A 1kW Switch Mode	£300
Farnell XA35/2T	PSU 0-35V 0-2A Twice Digital	£75
Farnell AP100-90	Power Supply 100V 90A	£900
Farnell LF1	Sine/Sq Oscillator 10Hz – 1MHz	£45
Racal 1991	Counter/Timer 160MHz 9 Digit	£150
Racal 2101	Counter 20GHz LED	£295
Racal 9300	True RMS Millivoltmeter 5Hz – 20MHz etc	£45
Racal 9300B	As 9300	£75
Solartron 7150/PLUS	6½ Digit DMM True RMS IEEE	£65/£75
Solatron 1253	Gain Phase Analyser 1mHz – 20kHz	£600
Solartron SI 1255	HF Frequency Response Analyser	POA
Tasakago TM035-2	PSU 0-35V 0-2A 2 Meters	£30
Thurlby PL320QMD	PSU 0-30V 0-2A Twice	£160 – £200
Thurlby TG210	Function Generator 0.002-2MHz TTL etc Kenwood Badged	£65

HP/Agilent HP 34401A Digital
Multimeter 6½ Digit £325 – £375

HP33120A	Function Generator 100 microHz – 15MHz	£350
HP53131A	Universal Counter 3GHz Boxed unused	£600
HP53131A	Universal Counter 225MHz	£350
Audio Precision	SYS2712 Audio Analyser – in original box	POA
Datron 4708	Autocal Multifunction Standard	POA
Druck DPI 515	Pressure Calibrator/Controller	£400
Datron 1081	Autocal Standards Multimeter	POA
ENI 325LA	RF Power Amplifier 250kHz – 150MHz 25W 50dB	POA
Keithley 228	Voltage/Current Source	POA
Time 9818	DC Current & Voltage Calibrator	POA

Marconi 2955B Radio
Communications Test Set – £800

Boom time for battery traction

Techno Talk

Mark Nelson

We're back with batteries this month, but batteries of a very different kind – specifically, traction batteries for cars, buses, trams and trains. Not the sort you are likely to use in electronics hobby projects, but they may well trickle down into broader fields, cross-fertilising into much smaller types. In any case, many of us may be driving electric cars soon and we'll definitely want to know what makes them tick.

Today's urgent need to reduce carbon emissions and energy costs means that traditional petrol and diesel-engined road vehicles are no longer flavour of the month. Fully electric cars, having zero tailpipe emissions are greener, cleaner and better for the environment. New technology is creating even more environmentally friendly batteries, as well as focussing research into more efficient, quieter electric vehicle (EV) motors.

Existing electrified railways and tramway systems will continue to employ their overhead wires or conductor rails to deliver power to the traction units, but even here battery power has a significant role to play. Not all main-line railways are electrified, and on lines that have not yet been 'wired', battery traction offers a far greener option than choking diesels. It also avoids the visual intrusion of the 'electrical Meccano' and 'wire knitting' used for suspending the overhead line.

On relatively short routes, battery-powered trains are a very viable option, with a ten-minute battery charge giving a range of 50 miles. The British company Vivarail makes battery-driven passenger trains that are under consideration for use on the Valley Lines in south Wales. These would run from Cardiff under overhead wires to the outskirts of the city, simultaneously recharging their batteries, before continuing on battery power to destinations including Rhymney and Merthyr Tydfil.

Visual impact reduction applies also in cities, which is the reason why trams on a new route extension in Birmingham will run partially on batteries. Overhead wires are taboo in certain historic areas of the city, which is why trams will use their on-board batteries to protect the visual aesthetics of Birmingham Town Hall (built in 1834), the 1879 Council House and historic buildings in Paradise Street.

Perils of prediction

Atom physicist and Nobel laureate Niels Bohr famously said, 'Prediction is very difficult, especially if it's about the future.' So far, people have rather assumed that electric cars, buses, trams and trains will charge their on-board batteries before and after making every journey, which may sound logical, but it may also prove to be highly inconvenient. Recharging batteries is not an instantaneous process, and in an ideal world it would help if batteries could also be topped-up along the way.

Wireless charging

Experiments are already underway in which wireless power transfer (WPT) methods are used to charge the batteries of electric vehicles while they are in motion. The system is not totally wireless; the electric conductors that transfer the power are laid beneath the road surface. For this reason, some proponents use the term 'contactless'. Magnetic resonance is currently the preferred WPT method, on account of its efficiency and its extended charging distance, but this may well change. After all, the first domestic radio receivers used inconvenient crystal detectors and headphones, but these were soon supplanted by valve receivers fitted with loudspeakers.

Is the concept feasible? Yes, and already in 2016 Berlin became the first capital city to introduce wirelessly charged electric buses in a one-year experiment. Four Solaris Urbino 12 electric vehicles operated on route 204, their 90 kilowatt-hour batteries being recharged on the move in a matter of minutes. This was a topping-up process, as the batteries were charged to full capacity overnight with cables at the bus depot. The charging was carried out with metal plates under the road and it was claimed that the wireless charging system was so well shielded that the electromagnetic radiation was lower than that of a conventional induction cooker in your kitchen. It must be conceded that this first implementation was not entirely successful, but practice makes perfect and the experiment provided valuable experience for further development. Meanwhile, electric trams using the same Primove technology are operating successfully around the world.

All's fine now?

Well not quite. Mass replacement of internal-combustion engines will require a vast number of traction batteries, together with the mining and extraction of huge amounts of lithium. This in itself carries high environmental costs, and the problems don't end there either, because lithium-ion batteries cannot be recharged indefinitely. Their useful life is between seven and ten years, and to quote *The Engineer* magazine, action must be taken now to avoid a potentially massive waste management problem in the near future. Research work led by the University of Birmingham in collaboration with Newcastle and Leicester universities estimates that the one million electric cars sold in 2017 alone will produce 250,000 metric tonnes (half a million cubic metres) of unprocessed battery waste at life-end.

Landfill and storage of spent batteries is not an option, as they are flammable and might release toxic materials into the environment. On the positive side, recycling spent batteries could present a massive opportunity, as the manufacture of replacement batteries requires access to strategic elements and critical materials for key components in electric vehicle manufacture. On that basis, recycling exhausted lithium-ion batteries could provide a valuable secondary source of materials.

There is also another option, as explained by Prof Andrew Abbott of Leicester University, who was involved in the research. This would be to use batteries that are no longer powerful enough for vehicles to perform less-demanding tasks, such as storing electricity from wind turbines and solar farms. The team also said that devising better ways to gauge the health of a battery would make it easier to assess whether they could be reused or repaired. But most important, he stressed, would be to develop recycling techniques similar to those for lead-acid batteries, in which 100% of the materials are recycled.

Win a Microchip SAM D20 Xplained Pro Evaluation Kit

Practical Electronics is offering its readers the chance to win a Microchip SAM D20 Xplained Pro Evaluation Kit (ATSAMD20-XPRO).

The SAM D20 Xplained Pro evaluation kit is a hardware platform to evaluate the ATSAMD20J18A microcontroller. The evaluation kit is designed for evaluation and prototyping with the SAM D20 Cortex-M0+ processor-based microcontrollers, which are ideal for a wide range of home automation, consumer, metering and industrial applications These devices offer 256KB of self-programmable flash, 32KB of SRAM and are capable of running at up to 48MHz.

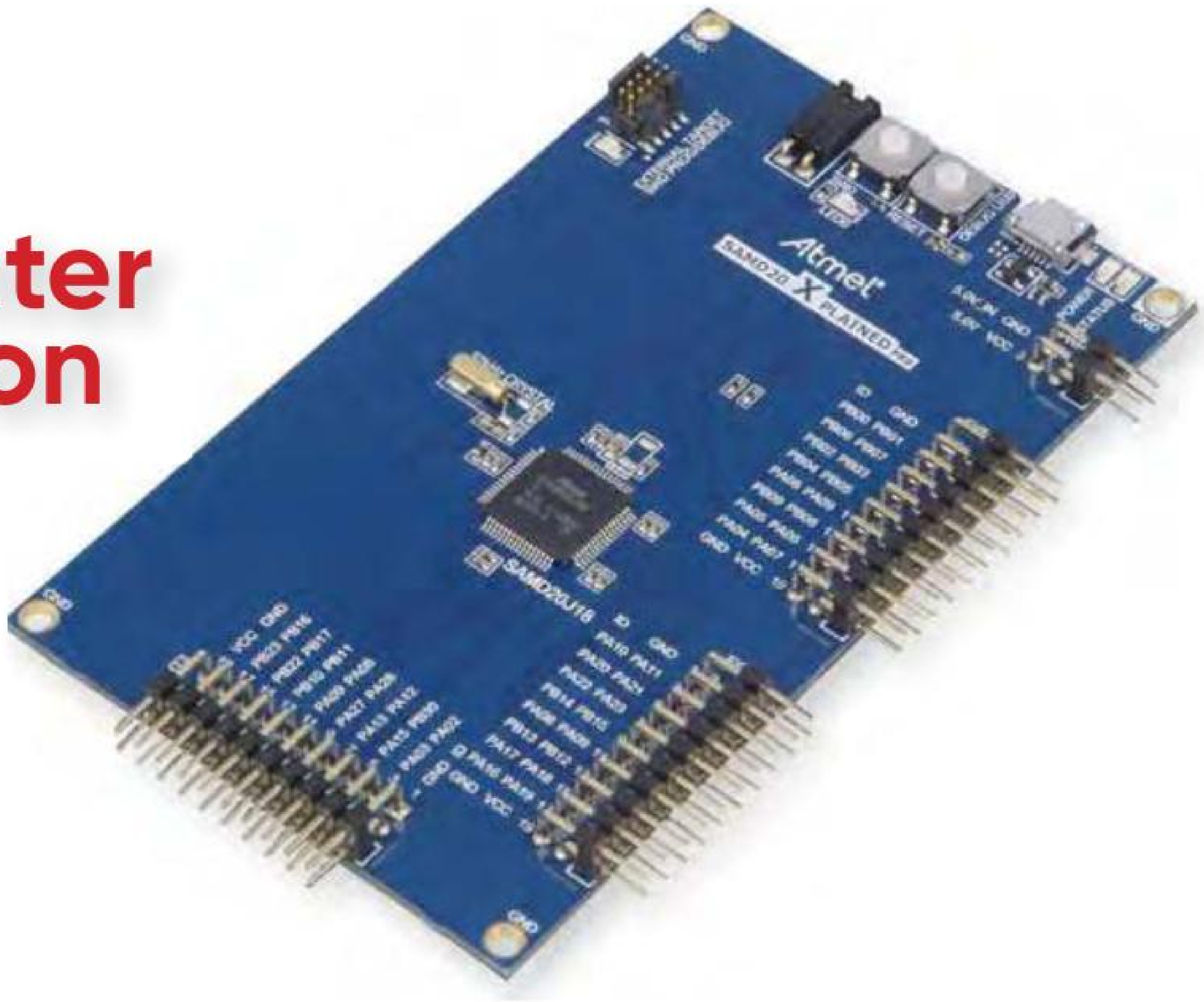
The Xplained Pro MCU series evaluation kits include an on-board embedded debugger, and no external tools are necessary to program or debug the ATSAMD20J18A. There are additional peripherals to extend the features of the board and ease the development of custom designs. The user can immediately start using the ATSAMD20J18A peripherals and get an understanding of how to integrate the device in their own design. Supported by the Atmel Studio integrated development platform, the kit provides easy access to the features of the ATSAMD20J18A.

Users also have the option to expand with the many Xplained Pro extension kits available, including the I/O1 Xplained Pro, OLED1 Xplained Pro, and PROTO1 Xplained Pro. The evaluation kit includes the following features:

- Atmel ATSAMD20J18A microcontroller
- Embedded debugger (EDBG)
- Digital I/O
- Two possible power sources
- 32kHz crystal

Worth $69.00 (approx £53) each

Free-to-enter competition



Microchip
SAM D20 Xplained Pro Evaluation Kit

How to enter

For your chance to win a Microchip SAM D20 Xplained Pro Evaluation Kit, enter your details in the online entry form at:

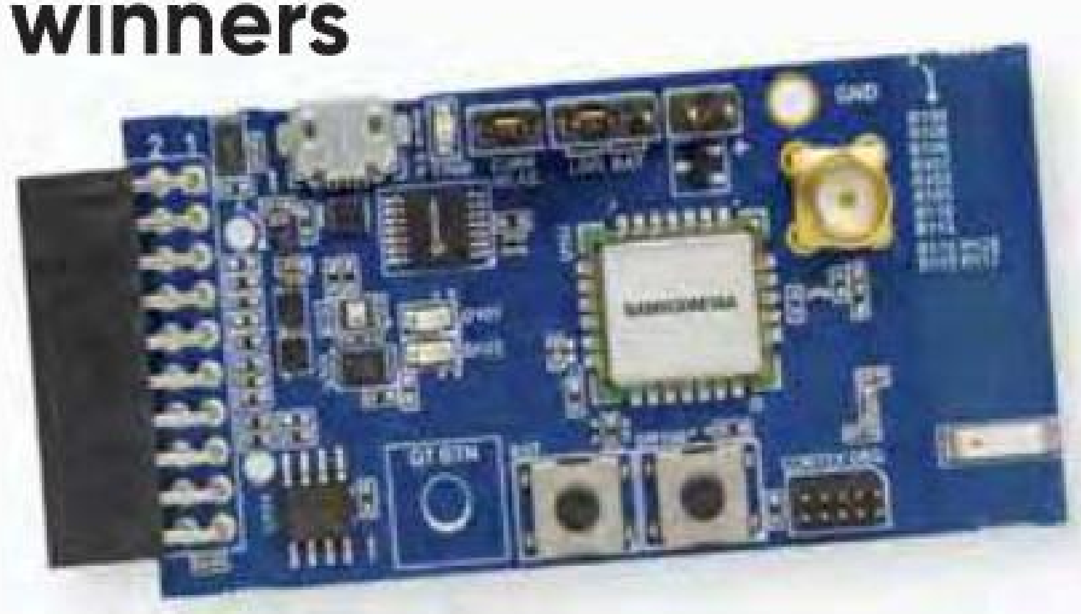
https://page.microchip.com/PE-D20-Kit.html

Closing date

The closing date for this offer is 29 February 2020.

November 2019 winners

Ed Rees
Andy Tyrrell



They won a Microchip SAM R30M Xplained Pro Evaluation Kit worth £31.25

Net Work

Alan Winstanley

This month, read the latest on Internet-based fraud, ID theft and advice on using 2FA (two-factor authentication) in your online dealings. Plus, SSDs – can you really entrust them with your digital data?

At the end of last year, the UK's National Fraud and Cyber Crime Reporting Centre – Action Fraud – reported the case of two brothers from the West Midlands who were involved in a nationwide eBay scam that raked in a six-figure profit over several years before they were finally caught. They recruited innocent so-called 'trading assistants' to sell on eBay what turned out to be non-existent mobile phones and tablets: the goods were never shipped but the monies were collected from the 'assistants' by the fraudsters themselves. After skipping bail and fleeing abroad, the crooks were extradited from The Philippines and were jailed for nine years for a scam that spanned nearly half a decade. The scale of the fraud became apparent when victims reported their losses to Action Fraud (**www.actionfraud.police.uk**), and local police units eventually joined the dots and caught the villains.

eBay, PayPal, Facebook, banks...

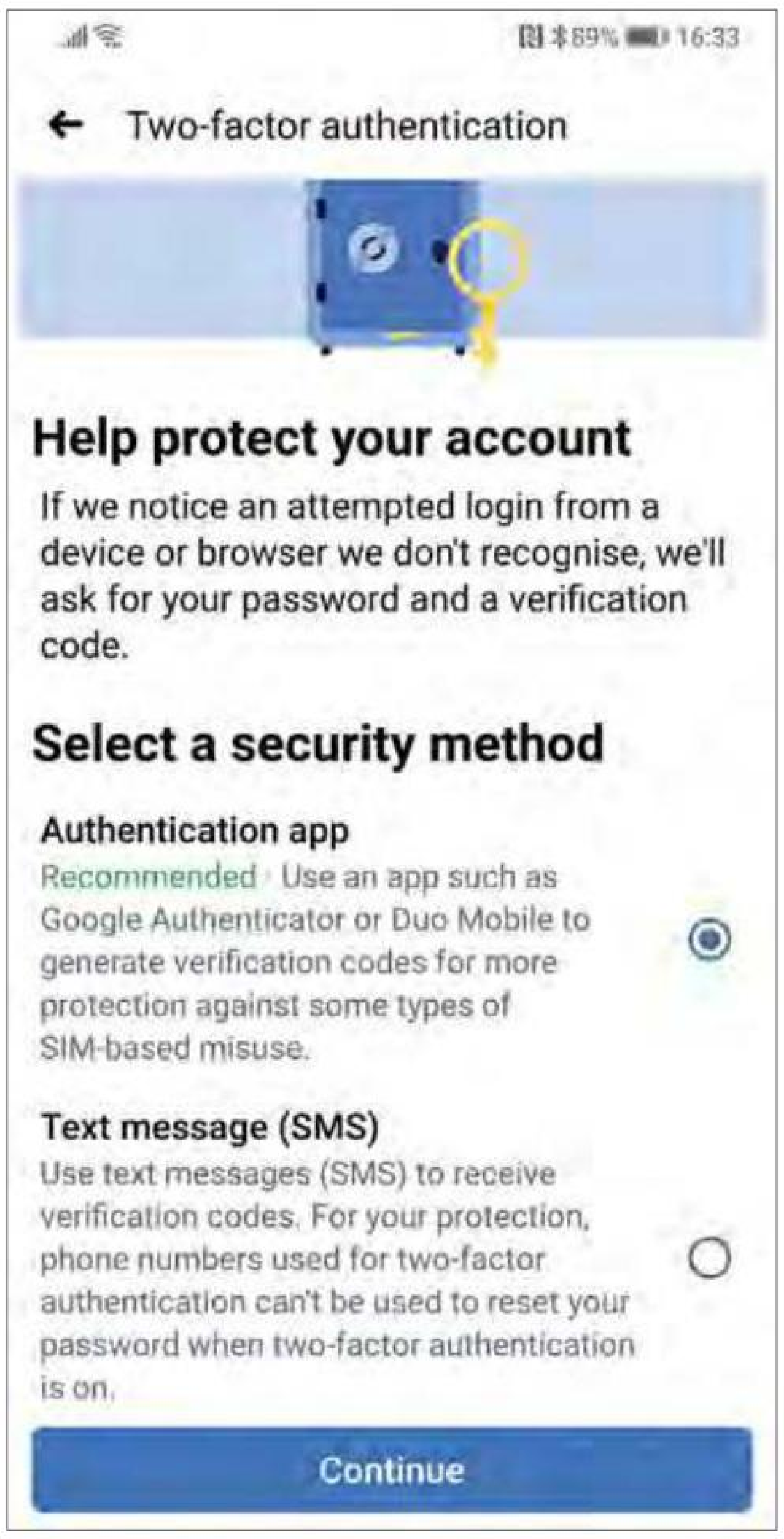
Meanwhile, their victims' eBay and Paypal accounts were frozen and some were hounded by debt collectors; one victim reportedly took out a second mortgage to raise cash, while another suffered serious bouts of depression. British legislation is very unforgiving about the plight of money mules – those who accept a 'fee' for passing bogus third-party transactions through their own accounts, whether driven by sheer desperation, naivety or (more often) plain greed. The penalties for this form of money laundering can be very harsh – up to 14 years imprisonment in the worst case, Financial Fraud Action UK reminds us.

In 2018, over 40,000 cases of bank account abuse due to suspected money mule activity were reported in the UK, says the fraud prevention service CIFAS. Social media and instant messaging channels were frequently used to attract mules before switching to secure encrypted channels like WhatsApp to ensnare them. Younger, impressionable people and university students were being taken in by these phony job offers. The obvious advice is to never, ever be tempted by job opportunities for 'sales agents', 'transaction processors' or any other schemes that let you 'earn money working from home' as a middle man, processing sales transactions through your personal accounts.

Too much information

Personal data is highly valuable to crooks and the author is sometimes shocked by the naivety of those who insist on sharing their details online when conducting private business in public. For example, I have just seen a rather elderly user post her full name, postal address, post (zip) code and hours of availability on a local Facebook group regarding someone taking away her old washing machine for scrap. (I asked admin to delete the post.) Useful snippets of data like these can help crooks to form a bigger picture about victims, leading to impersonation fraud or other forms of crime.

In the June 2018 CIFAS Report, *Wolves of the Internet: Where do fraudsters hunt for data online* (available at: **http://bit.ly/pe-mar20-wolves**) the value of personal data traded by ID thieves and fraudsters was laid bare. So-called 'Fullz' (a complete package of an individual's ID credentials) are traded openly both on the 'surface web' as well as the underground 'dark web'. Most prized on the dark web were PayPal logins, valued at £279 each, followed by eBay logins (£267). Online banking credentials were worth £161 but a lowly Facebook passcard was just £3.74, said the report. Nearly 50% of stolen data was then used to fraudulently apply for a credit card. Analysts Forensic Pathways Ltd in London used dark web crawlers to sample and collate stolen personal data over a 50-month period, and they estimated that someone's 'Fullz' data was worth about $42 and a BIN (Bank Identification Number) was about $21. A bank card's magnetic data 'dump' was more valuable, about $93. One person's Fullz profile was posted online where it attracted more than 270 hyperlinks to it, and it was used to apply for nearly two dozen dodgy accounts and cards.



Help protect your account

If we notice an attempted login from a device or browser we don't recognise, we'll ask for your password and a verification code.

Select a security method

Authentication app
Recommended · Use an app such as Google Authenticator or Duo Mobile to generate verification codes for more protection against some types of SIM-based misuse. ●

Text message (SMS)
Use text messages (SMS) to receive verification codes. For your protection, phone numbers used for two-factor authentication can't be used to reset your password when two-factor authentication is on. ○

Two-factor authentication (2FA) can alert users of unauthorised login attempts.

Two-factor authentication

As always, one should remain incredibly vigilant and safeguard against personal data theft as much as possible. In today's wired world, and with 5G coming around the corner, it's more important than ever to be on the look-out for fraud and ID theft and not ignore any suspicious events. As an extra safeguard, Two-factor authentication (2FA) adds another step to logging in by sending an SMS message and PIN code, for example, which can alert account holders if their logins have been compromised. The remote possibility of impersonation fraud means that 2FA may not be 100% foolproof though. To enable it, go to: My eBay -> Personal information -> Security information -> 2 Step verification. Gmail and Facebook users should sign up for 2FA if you haven't already done so – for further details see **http://bit.ly/pe-mar20-2fa** and **http://bit.ly/pe-mar20-2fa-fb**

Solid-state survivor

My thanks to regular *PE* reader Godfrey Manning who followed up on my January item concerning upgrading personal computers with solid-state drives (SSDs). Godfrey asks:

'How reliable have you found solid-state drives? I've had a couple fail after just a few months. The system freezes or displays a "Blue screen" error. For example, a full-system virus scan stalls or crashes. The drives are usually still readable by a stand-alone disc-copying utility, so I've reverted to SATA spinning discs. This has happened to different brands of SSD. After retrieving a Western Digital SSD, I broke it open and found it to contain a SanDisk chipset. Integral-brand SSDs have different chips.'

I admit to having a slight nagging doubt when installing my first solid-state drive at the heart of what is my main production machine. I chose to keep 'busy' data on my legacy spinning hard disk, while devoting the new SSD to hosting Windows. Many SSD users adopted that approach too, but others are not fazed by SSD reliability at all. I chose a Samsung Evo SSD mainly because Samsung is renowned for semiconductor memory and I preferred their Spinpoint hard drives in the past; I could also get one delivered quickly!

There is plenty of life left in spinning hard disks (a 14 TB drive is yours for £450) but the market has consolidated in recent years, taking some memorable brands with it: Samsung eventually sold its disk business to Seagate, the brand that also acquired Maxtor, while IBM hived off its Deskstar drive business to Hitachi, which in turn went to Western Digital. Flash memory chipmaker SanDisk is now owned by Western Digital too, which is why SanDisk-branded chips appeared in Godfrey's SSD. In a vote of confidence, *PE's* Internet Service Provider tells me that they replaced all their servers' hard drives with SSDs last year.

Amazon's Echo Flex is a smaller, plug-in adaptor unit offering Alexa services.

In my system, some local files are backed up online to Microsoft OneDrive, but I also back up key data to a RAID-style NAS using the excellent Macrium Reflect software (a free unsupported Lite version is available from: **www.macrium.com/reflectfree**). My Synology NAS currently hosts a pair of Western Digital 'Red' hard disks, but I also take a 'last gasp' ransomware insurance backup of it on a removable 2TB Samsung drive (which has a Seagate label underneath). Seagate has since dropped Samsung's branding in favour of using the Maxtor name on these drives, and SSD portable drives are catching on too.

As for the reliability of Flash memory, all one can do is hope for the best but insure against the worst. A new SanDisk MicroSD card in my new Christmas gift of a trail camera failed without warning, taking some video snaps of elusive wildlife with it. I guess nothing is 100% secure.

News roundup: ransomware, Echo, browsers and space

Major currency exchange company Travelex suffered a serious ransomware attack on New Year's Day, which caused their trading platform and internal networks to go down for several days, with traders reverting to pen and paper. It is reported that *Sodinokibi* malware was introduced via unpatched Pulse Secure VPN servers. No details of any ransom demands were publicised.

Amazon has launched a small plug-in version of its Echo smart speakers. The Echo Flex offers the usual Alexa voice control in a smaller form factor that fits directly into a wall socket. Amazon emphasises that Echo Flex is not optimised for music playback. It costs just £24.99, UK and Ireland only. More details at: **https://amzn.to/2urCyEB**

As an alternative to Chrome or Firefox, Microsoft is re-engineering its Edge desktop web browser to make life easier both for website users and developers. Edge is adopting the Chromium open-source model as its engine, which they hope will improve website compatibility and performance for surfers and encourage developers to adopt more uniform web design standards. Users can expect it to be delivered by a Windows 10 update and a new version of Edge for macOS is slated. More details from: **www.microsoftedgeinsider.com**

Blue Origin's New Shepard VTOL space vehicle lands successfully after a test launch.

In early January, SpaceX launched its third crop of 60 satellites for Starlink, its low-earth orbit Internet satellite project. The first stage successfully landed vertically again on their drone vessel 'Of Course I Still Love You' in the Atlantic (seen on the Youtube video **https://youtu.be/HwyXo6T7jC4** at 20:14). Astronomers are increasingly concerned about the impact that the satellite constellations are having on observations, but thousands more satellites are set to be launched in the future. Special satellite coatings are being tested that might minimise interference with astronomers' optical instruments. This year may also see test flights of SpaceX's new Starship, a 37-engined super-heavy lifter designed to carry crew or up to 100 tonnes of cargo, possibly to the Moon or Mars.

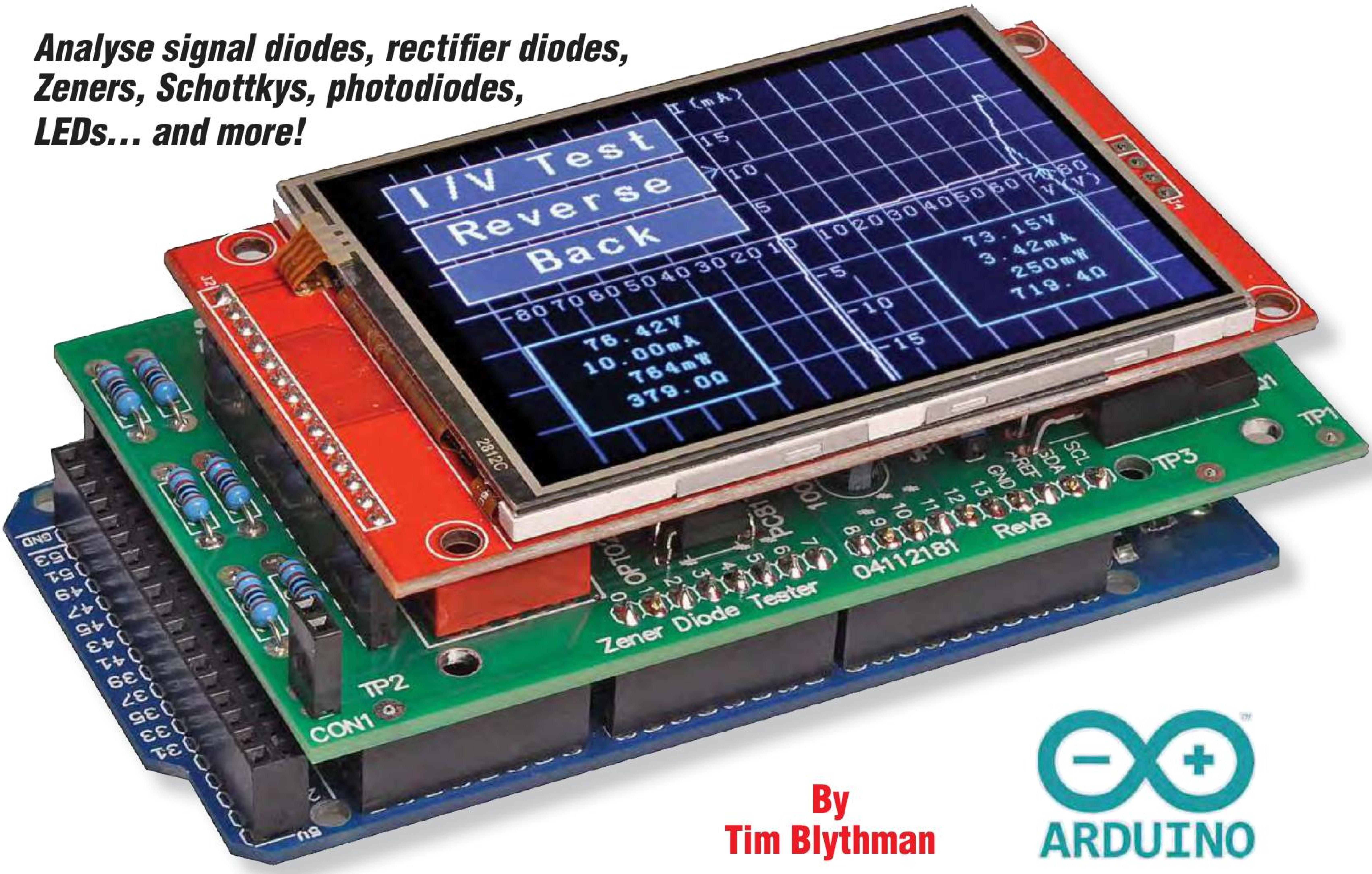
Another name to watch out for is Blue Origin, an aerospace firm founded by Jeff Bezos of Amazon fame. It has been quietly developing lower cost, reusable vertical take-off and landing sub-orbital vehicles that will take payloads – and maybe paying passengers – into space and back again (useful when you're a passenger). Space fans will enjoy Blue Origin's New Shepard vehicle test launch and vertical landing at: **https://youtu.be/9pillaOxGCo**

Unseasonably mild weather conditions recently affected some Freeview TV broadcasts for several days. The useful Radio & Television Investigation Service website is provided by the BBC to help troubleshoot problems with both terrestrial and satellite systems. The RTIS blamed the fine weather conditions for causing 'tropospheric ducting' – more details of this, along with lots more practical advice is on the RTIS website at: **http://bit.ly/pe-mar20-rtis**

That's all for this month's roundup – see you next month for more *Net Work*.

The author can be reached at:
alan@epemag.net

Analyse signal diodes, rectifier diodes, Zeners, Schottkys, photodiodes, LEDs… and more!



By Tim Blythman

Diode Curve Plotter

Our new *Diode Curve Plotter* is a superb piece of test gear that is both practical and educational. It's very versatile and fits in the palm of your hand. It automatically tests diodes in both directions and plots the resulting current/voltage curve on a colour LCD screen. It tests zener diodes up to about 100V, but it can also test LEDs, schottky diodes, regular diodes, transient voltage suppressors and more.

There have been many diode tester projects. Tyically, they were rather simple and only able to provide a measurement of, for example, a zener voltage. Also, they were often treated as an 'add on', and required a separate multimeter to display the result.

This new unit utilises the same 2.8-inch colour LCD touchscreen used in the *Micromite BackPack* from May 2017, but this time it's being paired with an Arduino Mega board and a custom PCB, which provides the user test interface.

What can it do?

The main feature of the *Diode Curve Plotter* is that it performs a full bidirectional current/voltage (I/V) sweep of a connected diode (or other component) and display the results in a graphical form.

For zener diodes, the zener voltage and current are displayed on the screen, along with the zener impedance at that point. You can move the test point to get different voltage, current and impedance readings along the curve.

The unit can produce up to 100V at up to 30mA for testing diodes, providing a wide testing range. For devices like LEDs, you can limit the test voltage and current to avoid damaging them during testing. It also has a specific LED testing mode, to make that job even easier.

The plot data can also optionally be sent to a connected computer as rows of CSV (comma-separated value) data, allowing plots to be stored and analysed further if necessary. You can plot and analyse this data on your PC using just about any spreadsheet program.

Getting back to the unit itself, cursors on its screen allow the operating point to be varied, by selecting either a voltage or current, so that the operating conditions can be examined across the range of the plot. For example, you could investigate how a zener diode performs at points away from the 'knee' of the zener curve.

The hardware scans the diode in both quadrants. It shows the full plot on the display, but only the forward operating point conditions are displayed in detail. A 'Reverse' button allows the plot to be flipped so that the reverse characteristics can be checked without rerunning the test. This is handy if the diode is connected backwards, or to check its behaviour in both forward and reverse directions.

The unit has adjustable current, voltage and power-limiting parameters. But given that each test takes a few seconds to complete, even if these limits are set slightly high, any over-current or over-voltage condition is quite brief and unlikely to cause any damage.

The LED test mode is essentially a constant-current mode, which provides a set output current. It shows the forward voltage, power and voltage/current ratio for the connected device.

The current and voltage limits can be set to the nearest milliamp and volt respectively, so even unknown devices can be probed without risk.

If a resistor is connected, the voltage/current ratio will, of course, correspond to the resistance, and thus the diode tester can even be used as a very basic ohmmeter.

How it works

The *Diode Curve Plotter* is a sandwich of three boards: an Arduino Mega or compatible board forms the bottom layer and provides the processing power. The LCD touch panel is the top layer, providing the display and user interface. In the middle, the custom PCB contains the other parts, which measure the parameters of the connected diode.

By the way, the reason we are using an Arduino Mega rather than an Arduino Uno in this project is that we need the extra Flash memory space provided by the larger chip on the Mega. We are not using any of the extra pins.

High-voltage generator

As we noted, the unit can test diodes up to 100V, but it runs from 5V DC, so it needs a way to generate higher voltages to apply to the device under test (DUT).

The circuit diagram of the *Diode Curve Plotter* is shown overleaf in Fig.1.

Inductor L1, N-channel MOSFET Q1 and diode D1 operate as a standard boost converter which is driven by IC2, an LM311N comparator.

Zener Diode Tester
Back
Standard Device
400mW (100mWtarget)
Down
Up
Previous
Next

The plotter mounted in a UB3 Jiffy Box, with a laser-cut front panel to reveal the touchscreen display.

Features and specifications

- **Tests zener diodes, LEDs, TVSs, silicon diodes, schottky diodes and more**
- **Colour touchscreen interface**
- **Tests up to 100V/30mA from 12V DC supply (can run from 5V, including USB)**
- **Automatically plots I/V curve in both quadrants**
- **Reads out current, voltage, power and impedance at any point in the curve**
- **Adjustable current/power limit for smaller devices with 0.4W and 1W presets**
- **Simple LED testing mode**
- **On-screen button to show reverse characteristics**
- **Based on an Arduino Mega with custom shield**

It runs from a 5V DC supply, which is convenient, because that means you use a USB power bank, USB charger or even a PC/laptop USB port. But note that it may draw more than 500mA when testing higher-voltage, higher-power devices, so a computer USB port may drop its bundle under these conditions. A 1A+ charger or battery bank is recommended.

The boost regulator draws power from the Arduino's VIN pin, which is connected directly to its DC power jack. In case the unit is powered via the USB socket instead, the 5V supply flows through schottky diode D2 into the VIN rail, thereby powering the boost converter.

IC2, the LM311 comparator, is used both as an oscillator to drive the gate of MOSFET Q1 and also as a current limiter. Note that an LM311 can only *sink* current at its pin 7 output, so pin D3 of the Arduino ('BOOSTCTL') must be high to enable the oscillator. This pulls pin 7 up via the 1kΩ resistor; normally, it is held low by a 10kΩ resistor, so Q1 is off by default.

Since the Arduino's D3 output is capable of generating a low-frequency PWM signal, we can switch the boost circuit on and off rapidly with a varying duty cycle to control the resulting boosted voltage.

The circuit around IC2 is not a fixed oscillator; instead, it monitors the current passing through inductor L1 using the 1Ω 1W series resistor. When Q1 switches on, the voltage across the 1Ω resistor increases as the current through L1 builds and its magnetic field builds up, until the threshold set by the comparator's resistor network is reached. At this point, Q1 is switched off.

The 100kΩ feedback resistor provides hysteresis, allowing the current to drop a small amount before Q1 switches on again and the cycle repeats. The resulting waveform has a high duty cycle, as required for a boost circuit with such a high output/input ratio.

When Q1 switches off, the voltage at the end of inductor L1 that's connected to its drain shoots up and so diode D1 becomes forward-biased, charging up the 1µF capacitor to a much higher voltage than the incoming supply.

This voltage is divided by 100kΩ/3kΩ resistors, filtered by a 100nF capacitor and fed to analogue input ADC1 of the Arduino. The divider provides a voltage which is within the 0-3.3V range of the Mega's analogue-to-digital converter (ADC).

While the Mega's ADC has a 0-5V range by default, we are using its onboard 3.3V regulator as a more precise reference, and so we can measure up to around 108V with this divider. Jumper JP1 usually feeds 3.3V into its AREF pin.

This divider also discharges the 1µF capacitor so that its charge will not persist after the boost converter is switched off. With the capacitor charged to 100V, the 100kΩ resistor dissipates around 100mW, which is well within the ratings of a small 1/4W resistor.

Note that the 13kΩ resistor connected to pin 2 of IC2 via a 10kΩ resistor sets the maximum inductor current (ie, in L1) which effectively determines the maximum voltage that the boost generator can produce, and also affects the maximum current that the unit will draw.

So if you want to reduce the maximum test voltage (eg, to allow the unit to run from a USB port that

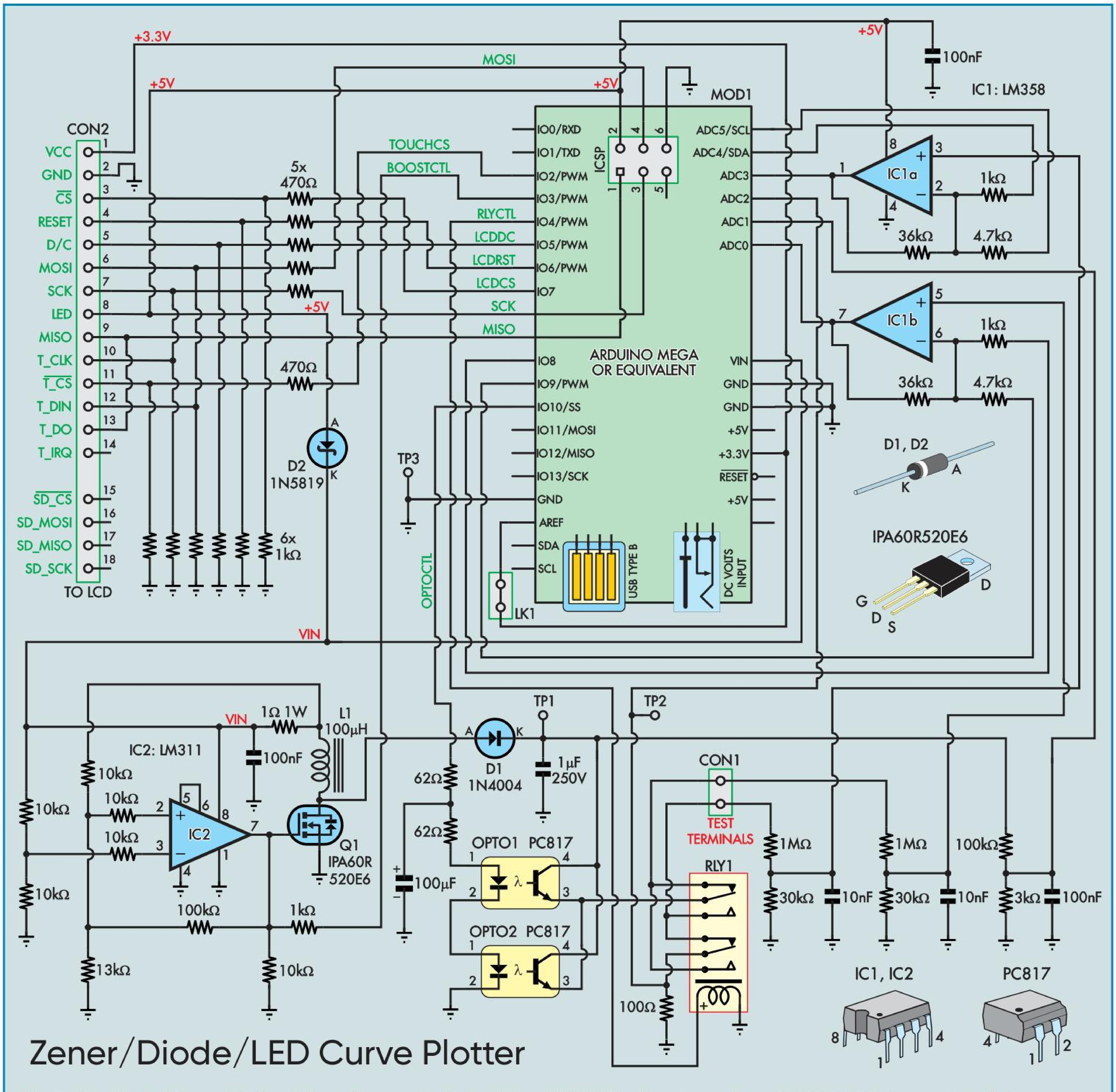

+3.3V
MOSI
+5V
+5V
+5V
100nF
MOD1
IC1: LM358
CON2
VCC
GND
CS
RESET
D/C
MOSI
SCK
LED
MISO
T_CLK
T_CS
T_DIN
T_DO
T_IRQ
SD_CS
SD_MOSI
SD_MISO
SD_SCK
TO LCD
5x 470Ω
470Ω
+5V
6x 1kΩ
TOUCHCS
BOOSTCTL
RLYCTL
LCDDC
LCDRST
LCDCS
SCK
MISO
IO0/RXD
IO1/TXD
IO2/PWM
IO3/PWM
IO4/PWM
IO5/PWM
IO6/PWM
IO7
IO8
IO9/PWM
IO10/SS
IO11/MOSI
IO12/MISO
IO13/SCK
GND
AREF
SDA
SCL
ICSP
ADC5/SCL
ADC4/SDA
ADC3
ADC2
ADC1
ADC0
VIN
GND
GND
+5V
+3.3V
RESET
+5V
ARDUINO MEGA OR EQUIVALENT
USB TYPE B
DC VOLTS INPUT
LK1
IC1a
IC1b
1kΩ
36kΩ
4.7kΩ
1kΩ
36kΩ
4.7kΩ
D1, D2
A
K
IPA60R520E6
G
D
S
D
D2 1N5819
TP3
OPTOCTL
VIN
IC2: LM311
VIN
1Ω 1W
L1 100µH
100nF
10kΩ
10kΩ
10kΩ
10kΩ
10kΩ
IC2
10kΩ
100kΩ
1kΩ
13kΩ
10kΩ
Q1 IPA60R 520E6
TP1
TP2
D1 1N4004
1µF 250V
62Ω
62Ω
100µF
OPTO1 PC817
OPTO2 PC817
CON1
TEST TERMINALS
RLY1
1MΩ
30kΩ
10nF
1MΩ
30kΩ
10nF
100kΩ
3kΩ
100nF
100Ω
IC1, IC2
PC817
Zener/Diode/LED Curve Plotter

Fig.1: the *Diode Curve Plotter* is based on an Arduino Mega (MOD1), a boost regulator (IC2/Q1/L1), two optoisolators which operate as a controlled current source (OPTO1 and OPTO2) and a relay to reverse connections to the DUT (RLY1). The test voltages and current are fed back to the Arduino so it can plot the curve and display measurements.

can only supply 500mA) then you can drop the value of this resistor to 12kΩ or even slightly lower, down to as little as 11kΩ.

Test circuitry

The test voltage is fed to the DUT via two optoisolators, OPTO1 and OPTO2, and relay RLY1. At the other end, the DUT is connected to ground via a 100Ω resistor.

OPTO1 and OPTO2 are configured as a controllable current source, with both collectors connected directly to the high-voltage supply and both emitters to the DUT. Their phototransistors are connected in parallel to enhance the amount of current they can supply to the DUT.

Their LEDs are connected in series, so that the effective current transfer ratio (CTR) is doubled. They are controlled by a PWM signal from pin D10 of the Arduino which is fed to the two 62Ω resistors. The 100µF capacitor smooths the PWM signal, in combination with those resistors, so that a steady, controllable current flows.

The modulated current goes to the DUT via relay RLY1. When its coil is energised, it reverses the connections to the DUT. The 100Ω resistor operates as a current shunt, allowing currents up to 33mA to be measured against the 3.3V reference voltage. The voltage across this shunt is monitored at the Arduino's A2 analogue input.

In practice, while testing a device, the unit sweeps the test voltage with the relay switched on, monitoring both the current and voltage, then performs another sweep with the relay switched off, so that current flows through the device in both directions during a single test pass.

Measuring circuitry

Four main parameters are measured by the Mega's internal 10-bit ADC. Two have already been mentioned: the voltage on the 1µF capacitor and the test current, as measured using the shunt. The other two parameters measured are the voltages at each end of the DUT. Both are fed into 1MΩ/30kΩ voltage dividers, giving the same 108V maximum reading.

These voltages are fed into the two halves of IC1, an LM358 op amp. By default, these are configured as unity-gain buffers, with the 36kΩ resistor in the feedback path having little effect. In this mode, voltages up to 108V can be measured with around 0.1V resolution (108V / 2^{10}).

But there is also a 4.7kΩ resistor and 1kΩ resistor from the inverting input of each op amp to two digital pins on the Arduino. These are initially left floating, and in this case, do not affect the op amp's operation.

However, if either is pulled low by its corresponding pin on the micro, that changes the op amp gain to either 8.66 (ie, 36kΩ÷4.7kΩ + 1) or 37 (ie, 36kΩ÷1kΩ + 1). This amplifies the sensed voltages, giving resolutions of around 10mV and around 3mV respectively, with the maximum readings being about 12.5V and 3V. So the gain is only increased when measuring lower voltages, to improve resolution.

All ADC measurements are sampled 16 times and averaged to improve precision and stability. Any error due to input offset will be taken care of during calibration stages.

Touchscreen interface

The touchscreen plugs into header socket CON2. The screen is powered from the 3.3V regulated supply, while the backlight is powered from the 5V rail. The Arduino controls it over two SPI (serial peripheral interface) buses. One is used for updating the screen and one for getting data from the touch sensor.

Their MISO and MOSI (data) and SCK (clock) lines are connected together to share the same set of hardware SPI pins on the Arduino, via its six-pin ICSP header. The screen and touch controller have separate chip select (CS) pins, at pins 3 and 11 on CON2, so the Arduino can select which one it is communicating with by pulling one of the two digital outputs (D7 or D2) low.

These five lines, plus the data/control line on pin 5 of CON2, have 1kΩ resistors connected from each pin to ground, plus 470Ω series resistors between the LCD pins and the Arduino. These form voltage dividers, reducing the 5V swing on the Arduino outputs to a 3.3V swing, to suit the LCD electronics. The MISO line is driven by the LCD so no level shifting is needed, as the Arduino will read 3.3V as a high level.

The remaining five pins on CON2, the interrupt request line from the touch controller (T_IRQ) and the four SPI control lines for the SD card socket are unused and so are left disconnected.

Software operation

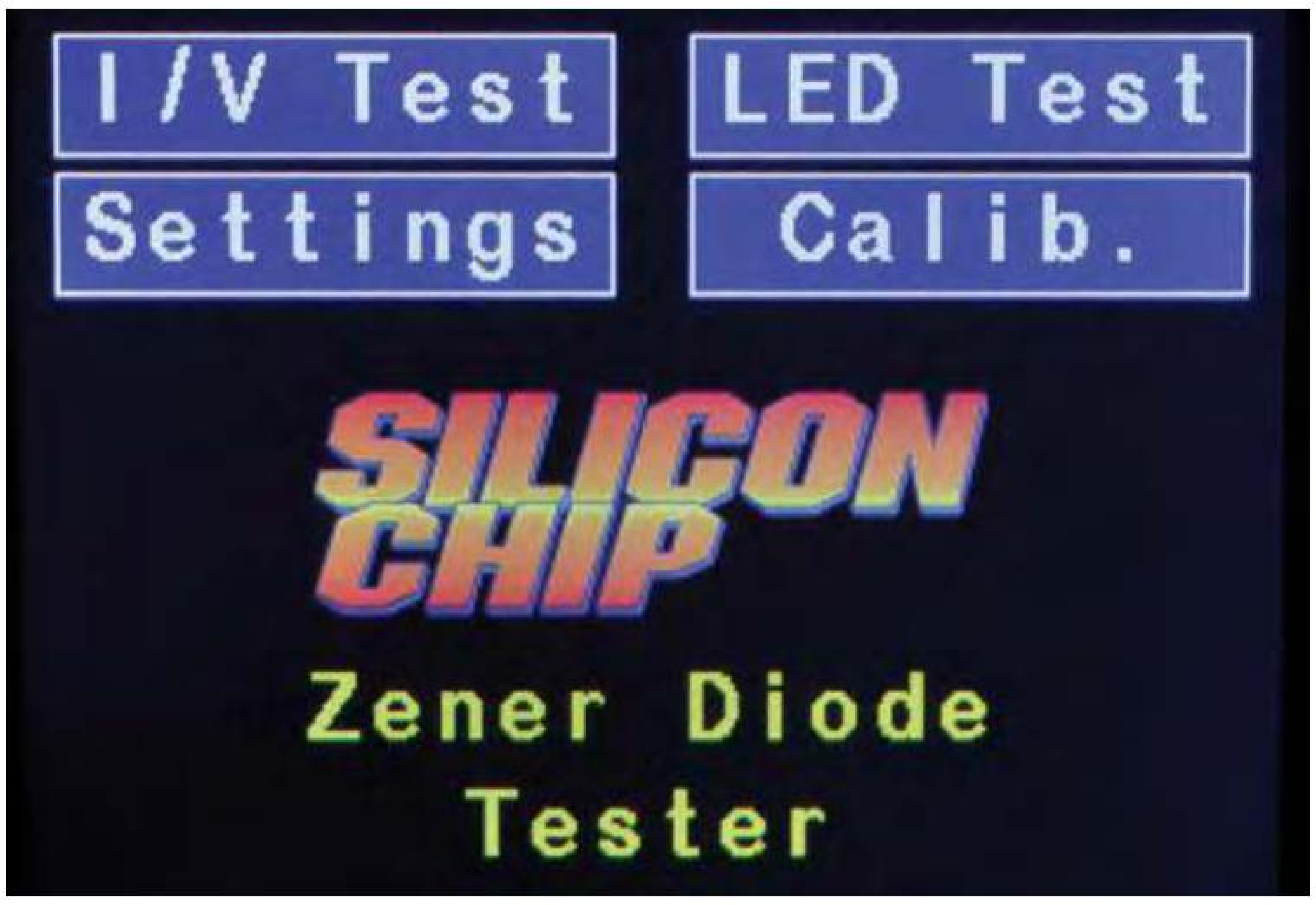
While testing, the boost converter is modulated by the PWM signal from pin D3 to maintain a voltage at TP1 that's slightly higher than the desired test voltage, but within the programmed limits. Then, the current through OPTO1/OPTO2 is varied across the testing range.

The difference between the voltages measured at both ends of the DUT, plus the current through the 100Ω shunt are recorded in an array.

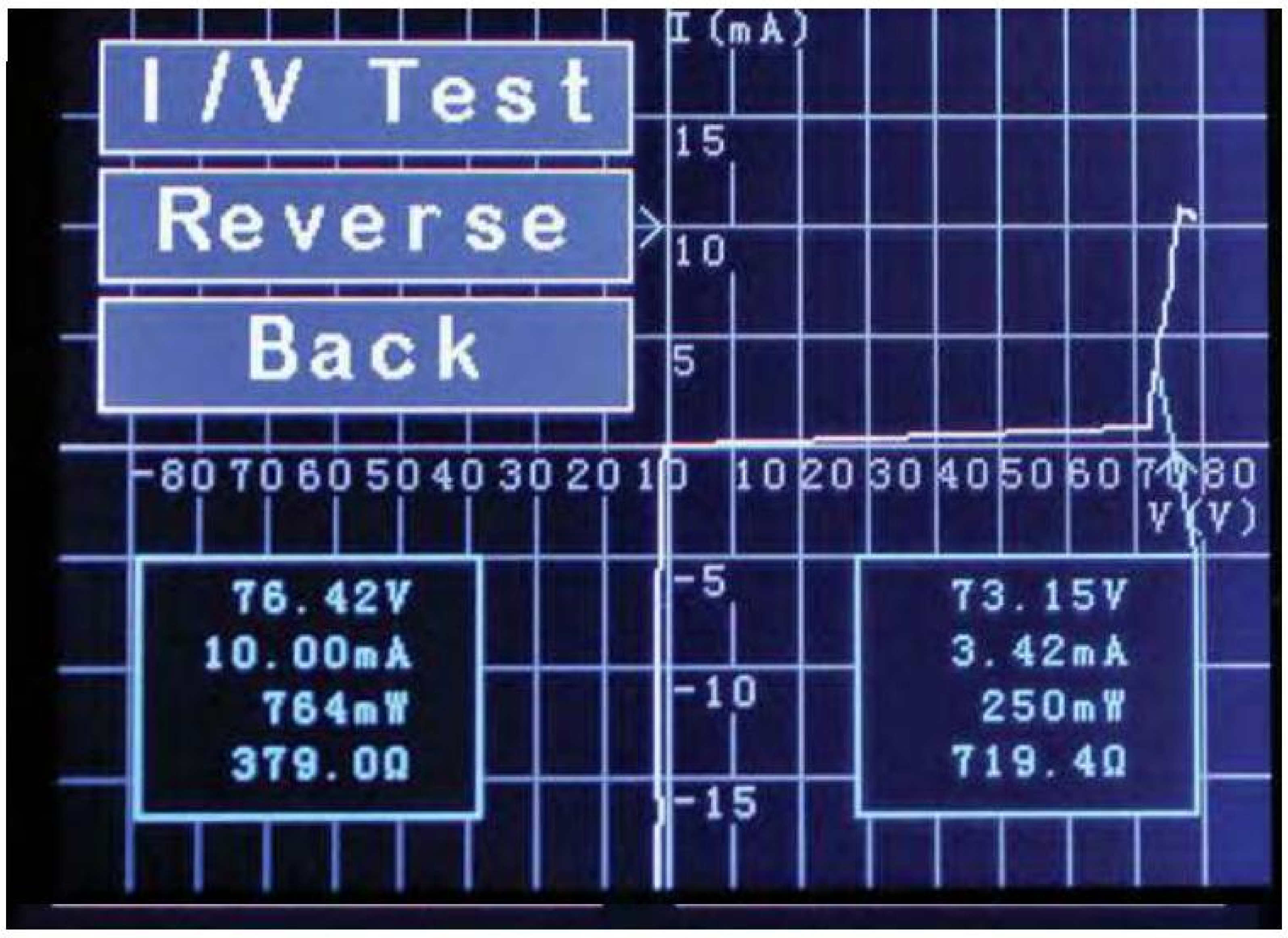
RLY1's state is toggled to reverse the polarity of the DUT, and the test is repeated, after which the results are plotted in a graph on the screen.

The Arduino Mega interpolates the data points to find the voltage and current at which the device power equals the selected operating point. The relevant figures are then shown in a small box on top of the graph, as well as drawing lines which show where that point is on the curve.

The box display includes the voltage, current and power at the operating point, as well as the zener impedance, derived from the gradient of the voltage/current curve at that point. The raw I/V data is then dumped to the serial port, where it can be read by the PC and used for additional analysis.



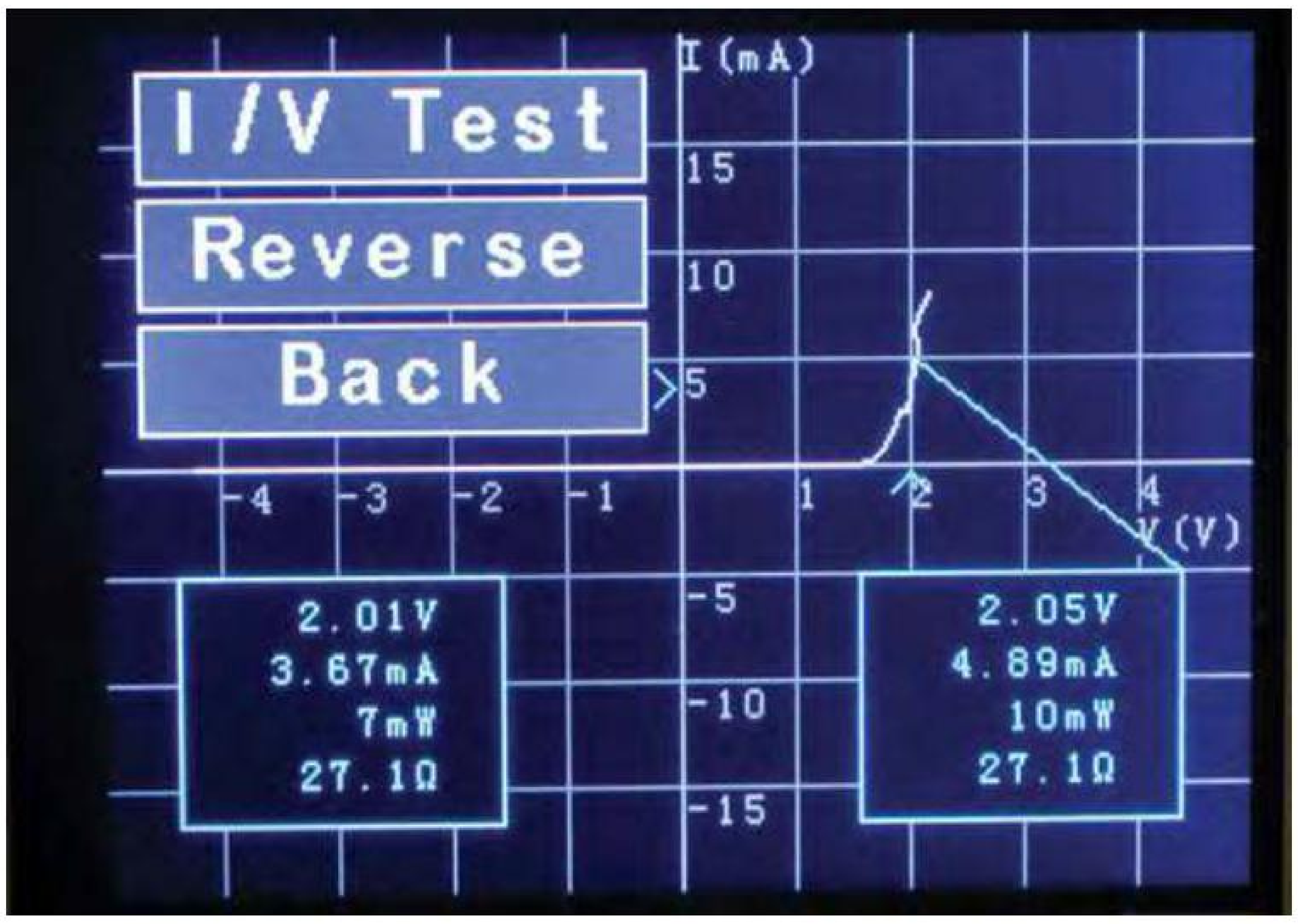
Screen1: the splash screen/main menu allows you to select between the two different types of tests (I/V Test or LED Test) and access the Settings and Calibration menus.



76.42V
10.00mA
764mW
379.0Ω

73.15V
3.42mA
250mW
719.4Ω

Screen2: the typical result of the I/V Test run on a 75V zener diode. A 250mW operating point is identified and indicated on the graph.



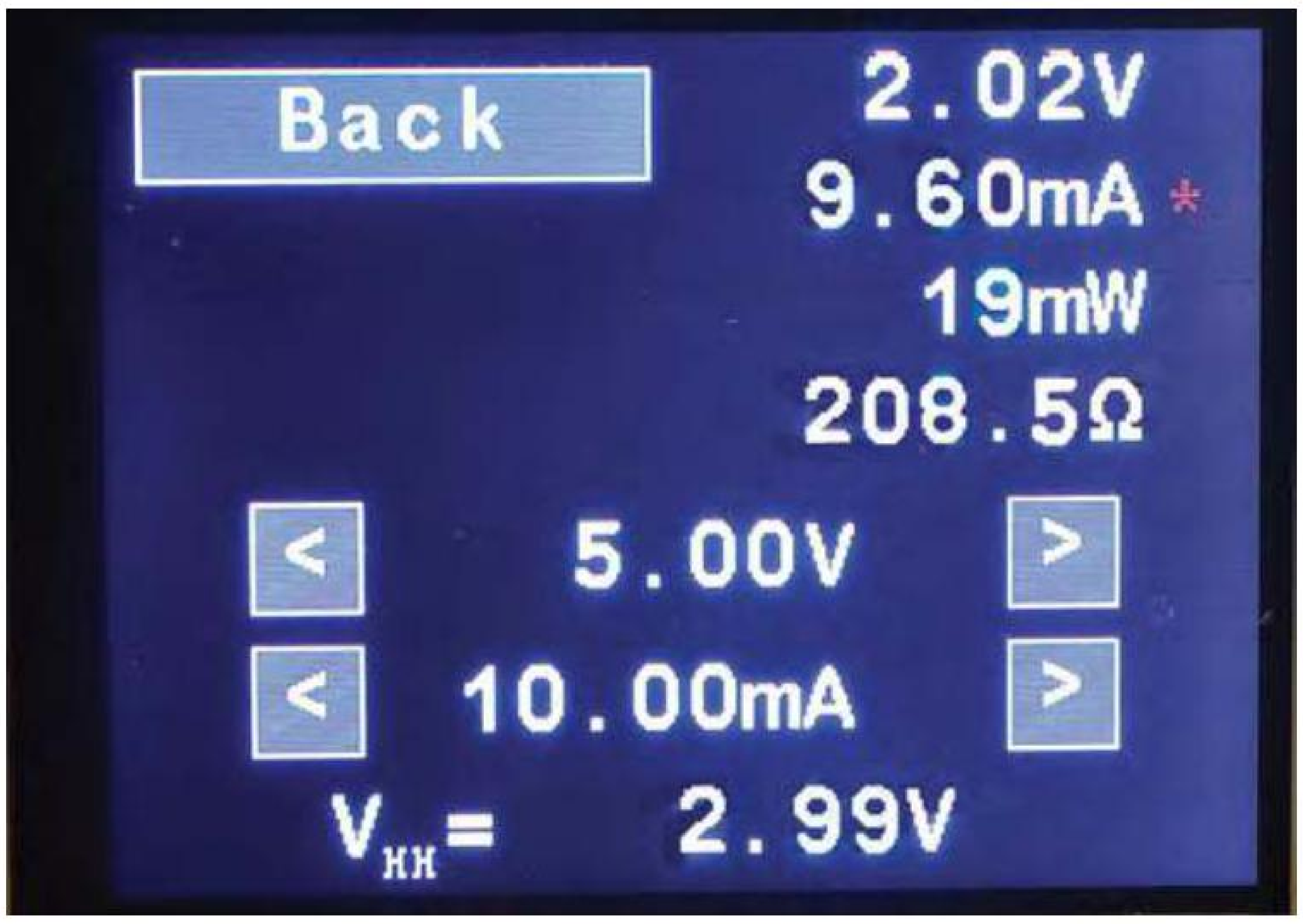
2.01V
3.67mA
7mW
27.1Ω

2.05V
4.89mA
10mW
27.1Ω

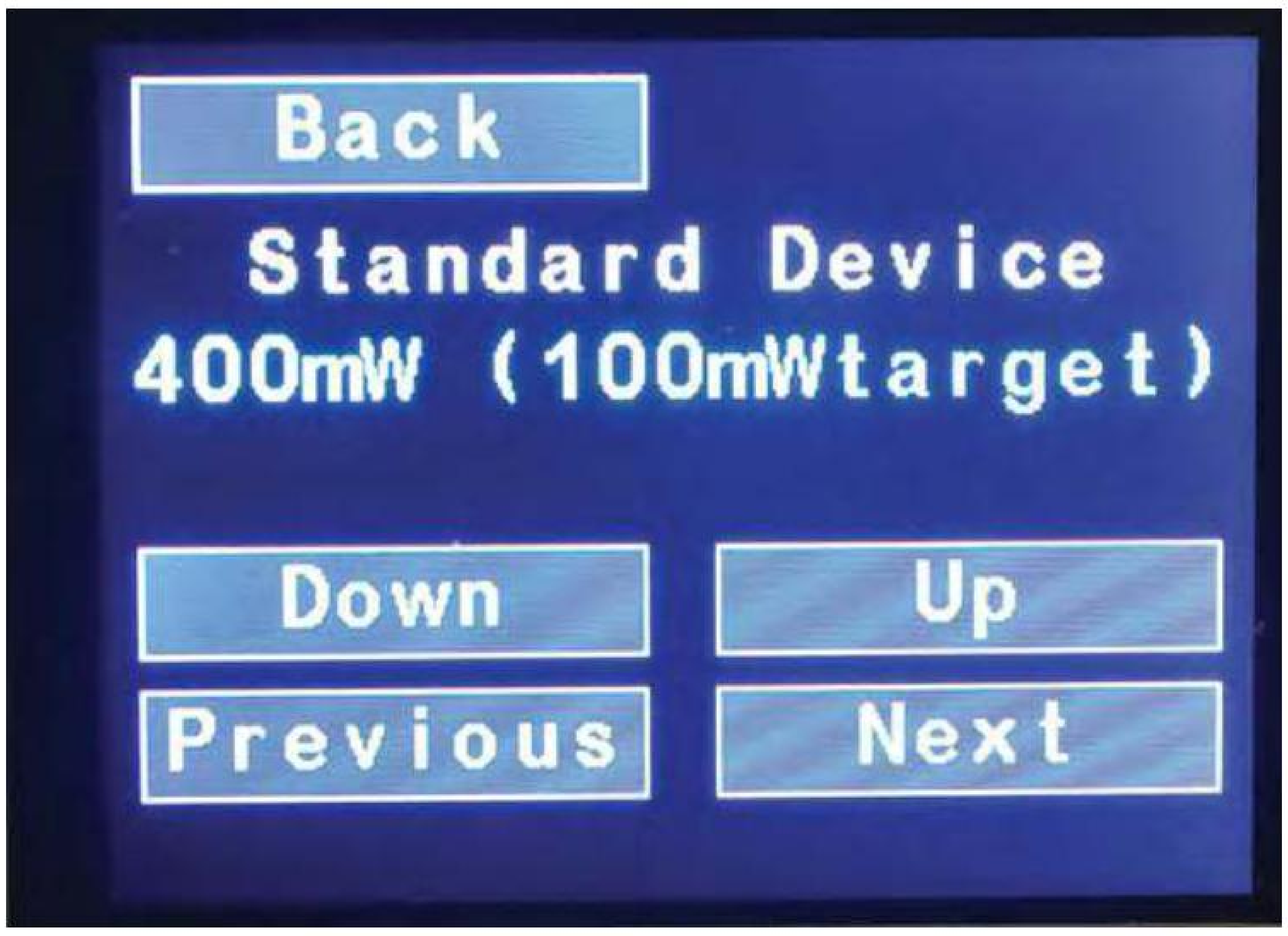
Screen3: here we have selected the I/V Test option with an LED connected to the unit and it has performed the measurements and plotted the graph. It's showing that 10mW is achieved, a forward voltage of 2.05V and a test current of 4.89mA.

The details of a second operating point can be analysed by touching the graph along the right-hand axis. If the graph is touched in the first quadrant (top right), then the current level is selected according to the vertical position of the touch.

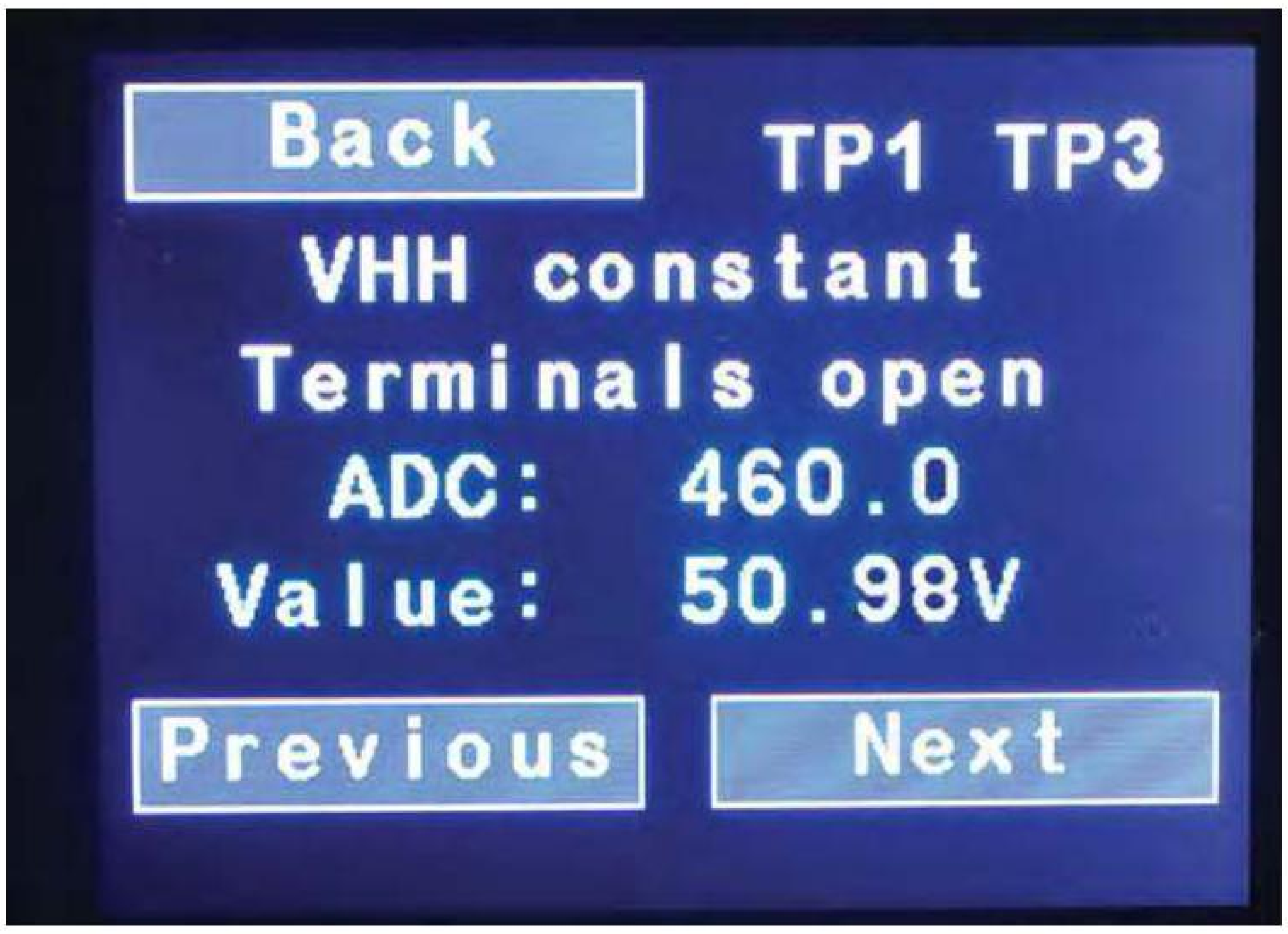
A second info box is displayed, showing conditions at the new operating point.


Back
2.02V
9.60mA *
19mW
208.5Ω
< 5.00V >
< 10.00mA >
V_{HH}= 2.99V

Screen4: in the LED test mode, where the forward voltage, current, power and zener impedance are continuously updated. You can adjust the maximum voltage and current applied to the LED directly with the arrows below.


Back
Standard Device
400mW (100mWtarget)
Down
Up
Previous
Next

Screen5: in the Settings screen, where you can select the type of device being tested, the maximum power and the target (nominal) power. The four buttons at the bottom change these values, then you press the Back button when finished.


Back
TP1 TP3
VHH constant
Terminals open
ADC: 460.0
Value: 50.98V
Previous
Next

Screen6: the Calibration screen reads out one of seven parameters, as measured by the Arduino, allowing you to compare them to readings made with a DMM and calculate coefficients to provide more accurate measurements.

Similarly, touching the graph in the fourth quadrant (bottom right) sets the voltage according to the horizontal touch position and displays a similar box.

The LED test page is much simpler, and in this mode tests are run continuously. The current is modulated to the limit set on that page, and the voltage is maintained within these limits by controlling the boost circuit.

With more screen space available, the statistics are shown in a larger font and they include voltage, current and power at the instant of measurement, as well as the ratio of voltage to current. This will not correspond to the zener impedance, but will be a fair measurement of the resistance of a fixed resistor.

Construction

All of the components mount on a shield PCB, which measures 99mm × 60mm, and is available from the *PE PCB Service* (coded 04112181). Use the overlay diagram in Fig.2 as a guide while building the board. Start by fitting the small (1/4W or 1/2W) resistors where shown. It's a good idea to measure the value of each one before fitting, as the colour bands can sometimes be ambiguous.

Solder diodes D1 and D2 in place next. They are different types and also oriented differently. Make sure that you don't mix them up and that the cathode stripes face in the directions shown in Fig.2. You can then fit the larger 1Ω 1W resistor.

Now install the seven capacitors, making sure that for the electrolytic types, the longer (+) lead goes into the pad marked with a '+' sign on the PCB overlay diagram and the PCB silkscreen printing.

Since the LCD stacks above this board, all components must project less than 12mm above the top surface of the PCB. If any of your electrolytic capacitors are 12mm high or taller, you will need to lay them over on their side when you fit them.

Note that we give two options for the 1µF capacitor, a polyester 'greencap' and an electrolytic type. While both should work, we prefer using the greencap, despite the fact that it needs to be installed with its leads bent over to keep it under 12mm high. Greencaps have better performance than electrolytics. But either should work, so it's up to you.

Fit the two ICs next. They are different types but come in the same package – don't get them mixed up. Fig.2 shows where they go and the correct orientation of each. Make sure the pin 1 notch or dot is facing as shown before soldering the pins. Then fit the two optoisolators, again taking care that their orientation is as shown.

Now mount MOSFET Q1. You will need to bend its legs 90° to allow the body of the MOSFET to sit flat. Before attaching to the board with a 6mm machine screw and nut, check whether that screw will foul the USB socket on your Mega board once the two are plugged together. If so, you will need to omit the mounting screw.

Once you've sorted that out, ensure the writing on the tab is facing upwards and then solder its leads. Telecom-style relay RLY1 is installed next, with its pin 1 stripe to the left as shown. Then fit inductor L1, which is not polarised, so its orientation is not critical. Ensure that it is not too tall when installed; it may need to be laid on its side to keep it under the 12mm limit. Then solder pin header JP1 in place.

The four SIL pin headers for connection to the Arduino can all be snapped from a single 40-pin header. The easiest way to mount them to the board is to plug them into the Mega board, then slot the shield PCB over the top to ensure that everything is square and flush before soldering them in place. Note that they are inserted through the bottom of the PCB and soldered on the top side.

Use the same technique to solder the 2×3 female header to the board; again, it is mounted on the underside.

Note that you could use a stackable header set, such as Jaycar's HM3208, rather than the standard pin headers specified. But that is likely to change the overall height of the unit, and it may no longer fit in the specified case.

Use a similar technique to fit the 14-way female header which connects the LCD to this PCB. It goes on top of the board. Plug it onto the LCD header, then mount the LCD on the shield board using three 12mm tapped spacers and six 6mm-long M3 machine screws – don't attach it in the upper-right corner; there is no spacer mounted near IC2. Then solder the header in place.

The 2-way female header is used for CON1, which connects to the device under test. We found this type of header ideal for this purpose, as most smaller component leads simply plug into the sockets. However, you could chassis-mount banana sockets instead, and wire them back to the pads for CON1.

You may wish to solder extension leads to the pins of CON1 before fitting it, to make the top of the socket level with the top of the LCD once assembly is complete. That allows it to project through the hole provided in the laser-cut lid. But if you do so, insulate the wires with short pieces of heatshrink tubing or similar, keeping in mind that there can be around 100V between them during operation.

That completes the assembly of the shield board. Once this is done, double check your soldering. Given that the board can generate over 100V, you don't want a small error on the PCB to feed that back into your computer. You may wish to use the *USB Port Protector* we described in May 2019.

Unplug the shield/LCD assembly from the Arduino Mega now, as it's best to keep the boards separate until the Mega has been programmed, especially if the Mega has previously been programmed for another project.

Connect the Mega to a computer using an appropriate USB cable; most Megas have USB Type-B full-size sockets so you will need a Type-A to Type-B cable.

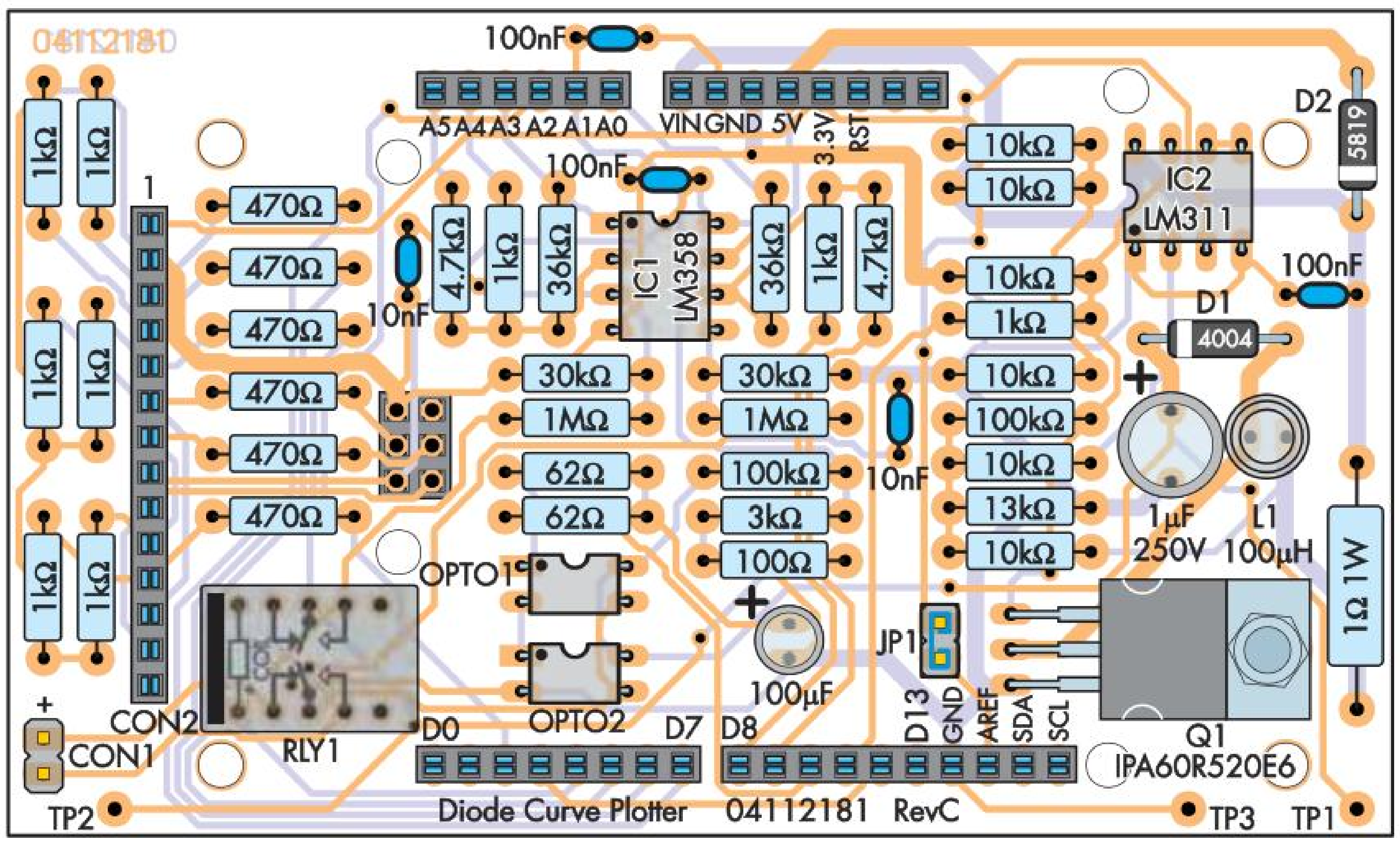
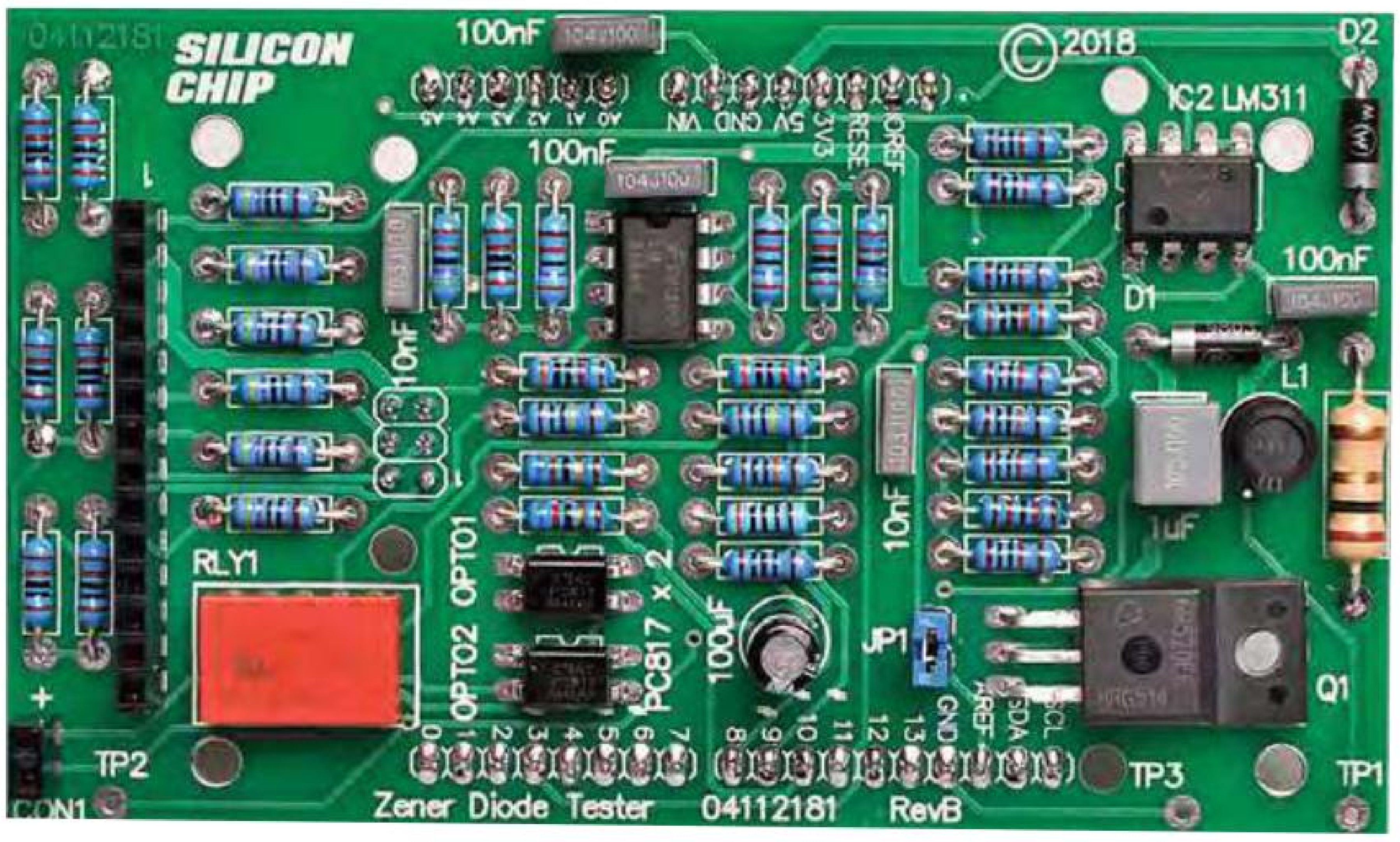


Fig.2: this component overlay echoes the silk-screen printing on the PCB surface shown below – between the two you should have no problems constructing the shield.



Installing the software

To install the software on the Mega, you need the Arduino IDE (integrated development environment) installed on your computer. The IDE includes a compiler and serial programming software, allowing the source code to be compiled and sent to the Arduino. The IDE can be downloaded from **www.arduino.cc/en/Main/Software**

We are using Arduino IDE version 1.8.5, but a newer version may be available by the time you check the download page.

Since we have written many of the libraries for this project ourselves, we have included all the necessary files in the sketch folder. Download the zip file from the March 2020 page of the *PE* website and extract the contents to a suitable location such as your 'Documents' or home folder.

Open the **Zener_Diode_Tester.ino** sketch file using the IDE. From the Tools menu, under Board, select 'Arduino/Genuino Mega or Mega 2560'. Then choose the appropriate serial port from the Tools -> Port menu. Click Upload or press Ctrl-U to start the compile and upload process. This may take a minute or two.

Unplug the USB cable and place the jumper shunt over the two pins of JP1 so that it is closed. Plug your shield PCB onto the Arduino Mega and then attach the LCD to the top of the PCB (if it isn't already attached). You are then ready for testing.

Testing and touchscreen calibration

Plug the USB cable back into the computer or if you have a 12V DC plugpack handy, use it instead. Ensure that the screen illuminates and displays the main menu page with the Silicon Chip logo. The Mega can have a start-up delay, so don't be alarmed if nothing happens for a few seconds. The sketch is written with a default touch panel calibration. Try pressing some buttons on the touchscreen and check that they respond as expected. If you find that they don't, or the touchscreen calibration seems inaccurate, or it is not responding to touch at all, you will need to use our provided calibration sketch to calculate new touch panel calibration parameters.

By the way, we've seen some 2.8-inch touchscreens which look more or less identical to others but the touch panel axes are reversed. If you have one of those, you will definitely need to go through the calibration process.

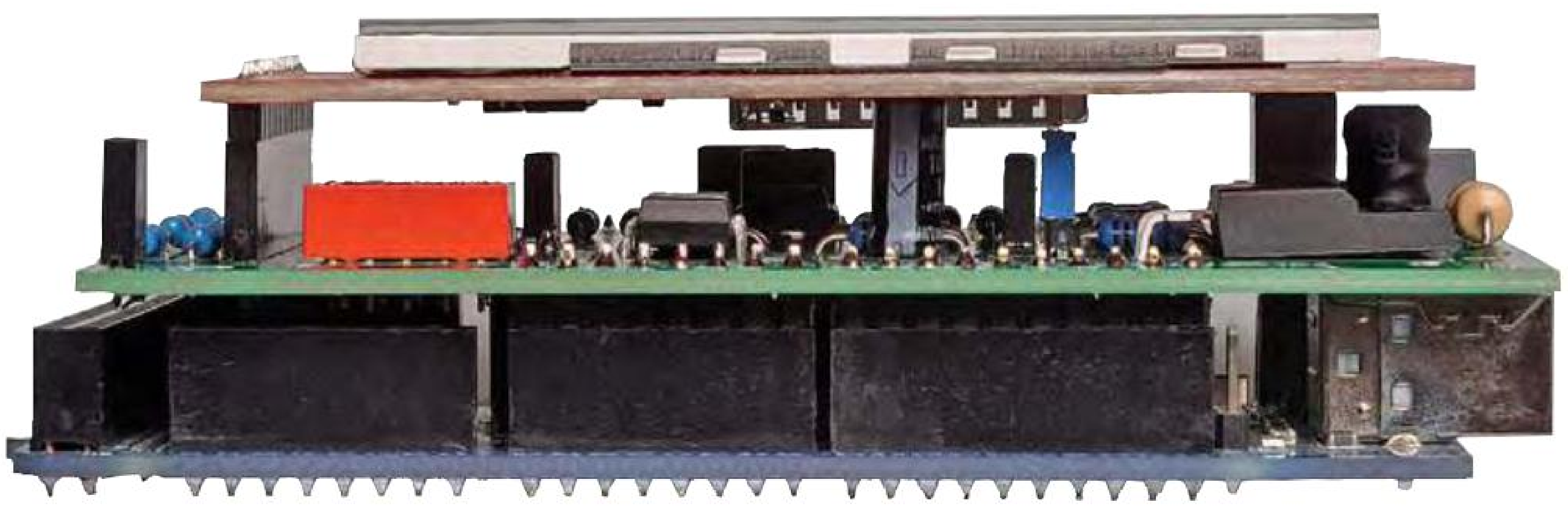
To do this, open the **AVR_LCD_BackPack_Touch_Calibration.ino** sketch and upload it to the Mega using the instructions above. Open the Serial Monitor from the Tools menu (or by pressing Ctrl-Shift-M) and set the baud rate to 115,200.

Following the instructions on the screen, use '1' (followed by Enter to send the command) to perform the calibrations. Then use '2' to test that the new calibration is accurate. Copy the new calibration constants from the Serial Monitor to the clipboard, as shown in Fig.3.

Now re-open the original **Zener_Diode_Tester.ino** sketch and open the **backpack.h** tab. Find the lines in the code which are shown in Fig.4. Replace the existing calibration constants with the new values you copied earlier, save the updated sketch and then upload it to the Mega. You are now ready for final testing.

Checking voltages and currents

Note as you read the following, that if you are using a computer USB port to power the unit during testing and calibration, some of the voltages mentioned below may not reach 100V and the unit may reset, due to the limited current capabilities of that port.



This photo, along with the one opposite, shows how the three boards are 'sandwiched' together – the Arduino Mega board on the bottom; the new shield board in the middle (green) and the 2.8-inch LCD touchscreen on top. It is designed to fit in a UB3 Jiffy box with a new laser-cut Acrylic lid. Note the connectors on the Mega board in the photo opposite – the USB on the left, and the DC power input at right.

Start by clicking the Calibration button and use the Previous and Next buttons to scroll through the various items.

In the top-right corner, the display shows where to connect your multimeter test leads to read the appropriate voltage. The second line indicates the name of the value being tested. The third line indicates whether the test terminals should be open or short-circuited.

They should be short-circuited for the current test (eg, using your DMM in current measurement mode); otherwise, there is no path for the current to flow.

The first item to be checked is the output of the high-voltage generator, and this should be up around 100V, with an ADC value in the 900s. If this is the case, the high-voltage system is working. If not, check (after powering off the unit) for wiring faults around the right-hand edge of the board, particularly around Q1 and L1.

The next three items measure the voltage at the positive test terminal with various gain settings. Using a DMM, measure the voltage at the positive terminal of CON1 (shown in Fig.2 and on the PCB) relative to TP3 (GND) and check that you get a reading that's close to the one shown on the screen.

The three following items are the negative test terminal voltage at its three different gain settings. Use the same technique as above to compare your readings to those shown on-screen. Any significant deviation in these voltages from reality indicates a problem in the vicinity of IC1.

The final item to check is the current reading, and as noted, it will only work if there is a path for the current between the test terminals. Switch your DMM to current (mA) mode and connect it across CON1. The displayed current should be around 30mA, with an ADC reading around 900. Compare this to the reading on your DMM. It should be close. If you don't get any current reading or it is wildly off, then you may have a problem with the circuit around the optoisolators.

To calibrate the unit, step through each reading and record the ADC value shown and an accurate measurement of the voltage (or current) using your DMM. Then divide the voltage or current value by the ADC reading, and write this value down.

The unit is then calibrated by modifying the scaling values in the sketch itself. This part of the code is shown in Fig.5. It's near the top of the file. Find those lines and change the values to those you wrote down.

If the values you have are significantly different from the defaults, you may have a problem with your board, or you might have made a mistake in calculating these values.

Performing this calibration adjusts the software to be accurate with the particular components on your board (eg, the exact resistor values).

After the values have been edited, the sketch will need to be uploaded again, as per the earlier instructions, to allow the new values to take effect.

Completing assembly

Once you are satisfied that the unit is calibrated and working correctly, it can be fitted in its case. Start by removing the screws holding the LCD screen onto the tapped spacers, then temporarily unplug the screen.

Now plug the shield into the Arduino Mega and secure the two together using the specified nylon machine screws and nuts, through the mounting holes near Q1 (adjacent to the Arduino SCL pin) and near the Arduino A5 pin.

Next, plug the LCD back into the shield but don't attach it using screws just yet. Slot the laser-cut lid panel over the LCD screen, then feed 10mm machine screws through

Parts list – Arduino-based Multi Diode Tester

1 double-sided PCB coded 04112181, 99mm x 60mm (available from the PE PCB Service)
1 Arduino Mega R3 board or equivalent [eg, Jaycar XC4420, Altronics Z6241]
1 2.8-inch LCD touchscreen [ILI9341-based touchscreen LCD panel, 320 x 240]
1 UB3 Jiffy box (included lid not required)
1 3mm laser cut Acrylic lid [Silicon Chip Online Shop Cat SC4927]
1 2-pin female header (CON1)
1 14-way female header (CON2)
1 2-pin header with jumper shunt (JP1)
1 6-pin, 2 8-pin and 1 10-pin header (to connect to Arduino)
1 2x3-pin female header (to connect to Arduino ICSP header)
1 DPDT relay with 5V DC coil and 250VAC-rated contacts, DIP-10 (RLY1)
1 12V 1A (or higher) plugpack with centre positive 2.1mm tip
1 100µH bobbin type inductor (L1)
2 M3 x 20mm nylon panhead machine screws
3 M3 x 12mm tapped nylon spacers
4 M3 x 10mm panhead machine screws
7 M3 x 6mm panhead machine screws
2 M3 nylon hex nuts
2 M3 hex nuts

WARNING

Do not touch any component leads while the unit is operating. 100V is enough to give you a bite!

Semiconductors
1 1N4004 400V 1A diode (D1)
1 1N5819 schottky diode (D2)
1 LM358 op amp, DIP-8 (IC1)
1 LM311N high-speed comparator, DIP-8 (IC2)
2 PC817 optoisolators, DIP-4 (OPTO1,OPTO2)
1 IPA60R520E6 700V N-Channel MOSFET (Q1) [mouser.co.uk / digikey.co.uk]

Capacitors
1 100µF 10V electrolytic
1 1µF 450V electrolytic ***or*** 250V polyester 'greencap'
3 100nF MKT or ceramic
2 10nF MKT or ceramic

Resistors (all 1/4W 1% unless otherwise stated)
2 1MΩ 2 100kΩ 2 36kΩ 2 30kΩ
1 13k 6 10kΩ 2 4.7kΩ 1 3kΩ
9 1kΩ 6 470Ω 1 100Ω 2 62Ω
1 1Ω 1W 5%

the panel and LCD, into the three tapped spacers below. Use the fourth 10mm machine screw and single nut to hold the lid onto the LCD screen using the remaining mounting hole, in the upper-right corner.

If you aren't planning to read measurement data out to a computer via the serial port and you are using a plugpack supply, then you only need to make a hole in the box base for the DC power jack. Otherwise, you will also need to make a cut-out to access the USB socket.

Make the holes in the lower half of the UB3 case using the drilling diagram (downloadable from March 2020 page of the *PE* website) as a guide. Finally, attach the lid to the top of the box using the supplied self-tapping screws.

Using it

This device will test just about any type of diode, including standard silicon diodes, schottky diodes, LEDs and unidirectional or bidirectional transient voltage suppressors.

Having connected the device to both of the test terminals, press the I/V Test button on the screen.

You should hear two clicks and the graph will be displayed. If you have inserted the component backwards, press Reverse to swap the plot around.

You can also touch the graph on the touchscreen to display figures for various voltages and currents. Cursors appear to show the point being touched and the relevant information is displayed in a second box on the bottom left of the screen.

Pressing Back returns to the main menu page. From there, press the LED Test button to start the LED test.

The voltage and current limits are set using arrow buttons at the bottom of the screen. These are soft limits which are controlled by the microcontroller, so the readings may occasionally drift above these settings. If this occurs, a small red asterisk is shown to alert you to that fact.

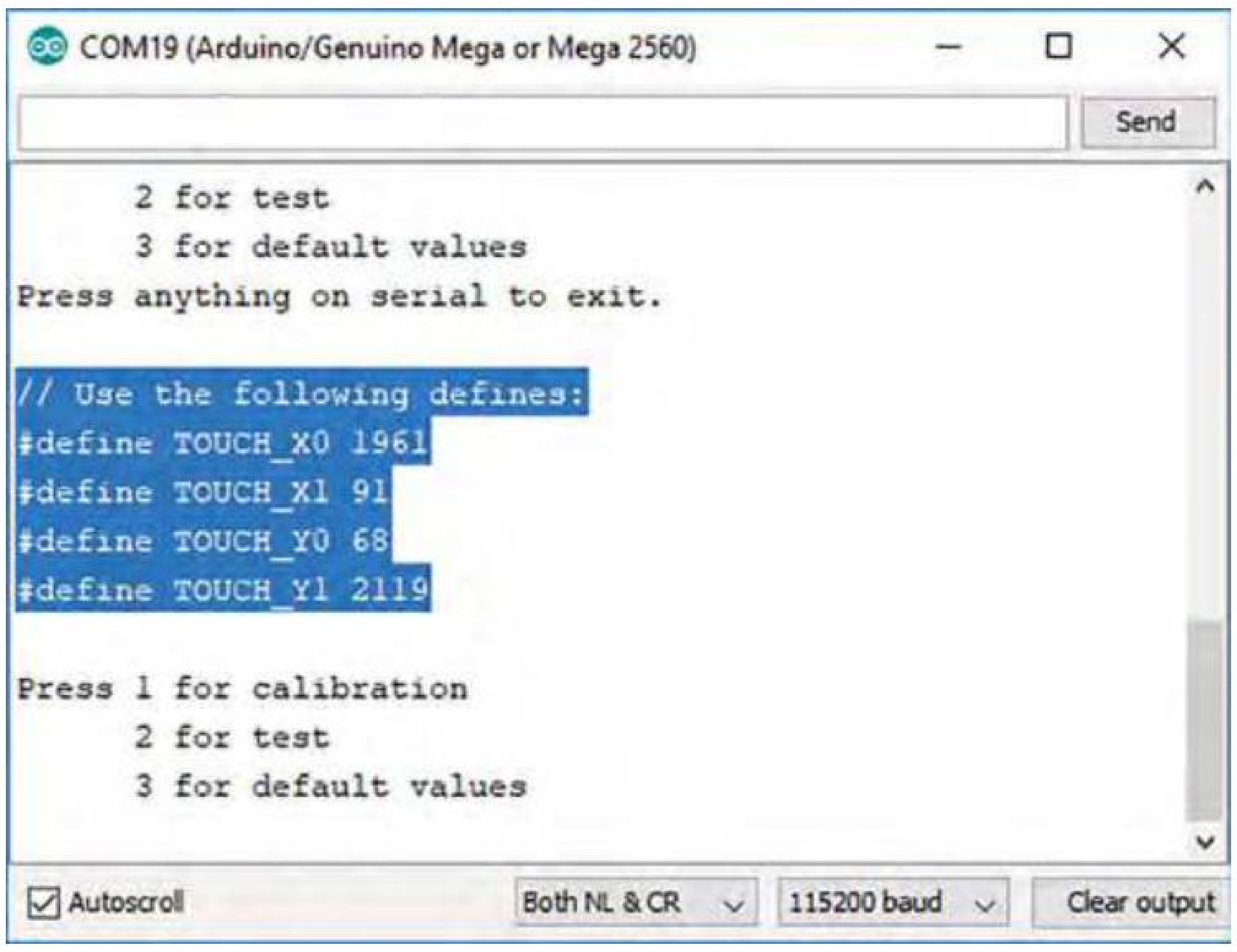
The high-voltage rail value is shown at the bottom of the screen, and the current device operating conditions are shown along the right-hand side of the screen.

Press Back again and then press Settings to go to the settings page. This sets the various parameters for the I/V Test mode (the LED test mode has its settings shown on that screen, as explained above). The Previous and Next buttons scroll between various items, while the Up and Down buttons change the values of those items.

There are seven settings available. The first allows you to select either a 400mW or 1W zener; it automatically sets the maximum and target power settings. If this is set to 'Manual' instead, the next two items can be used to set the maximum and target power manually.

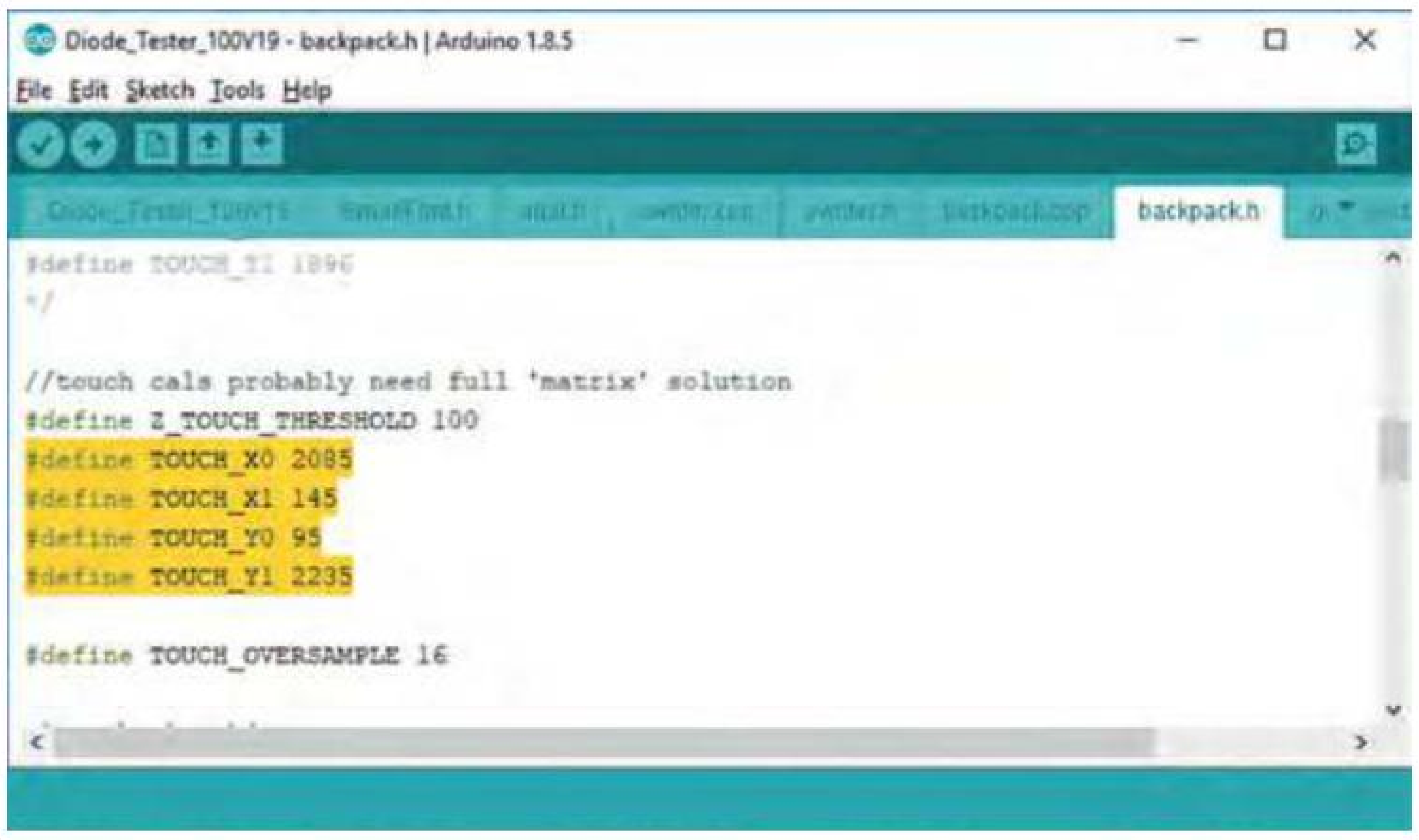
The following two items allow you to manually set a conservative current and voltage limit for I/V tests. When running I/V tests, the test is stopped if either of these limits is exceeded.

The final two items set the scale of the graph. If, for example, the voltage scale is set to 10V, then the horizontal axis of the graph will span –10V to 10V.



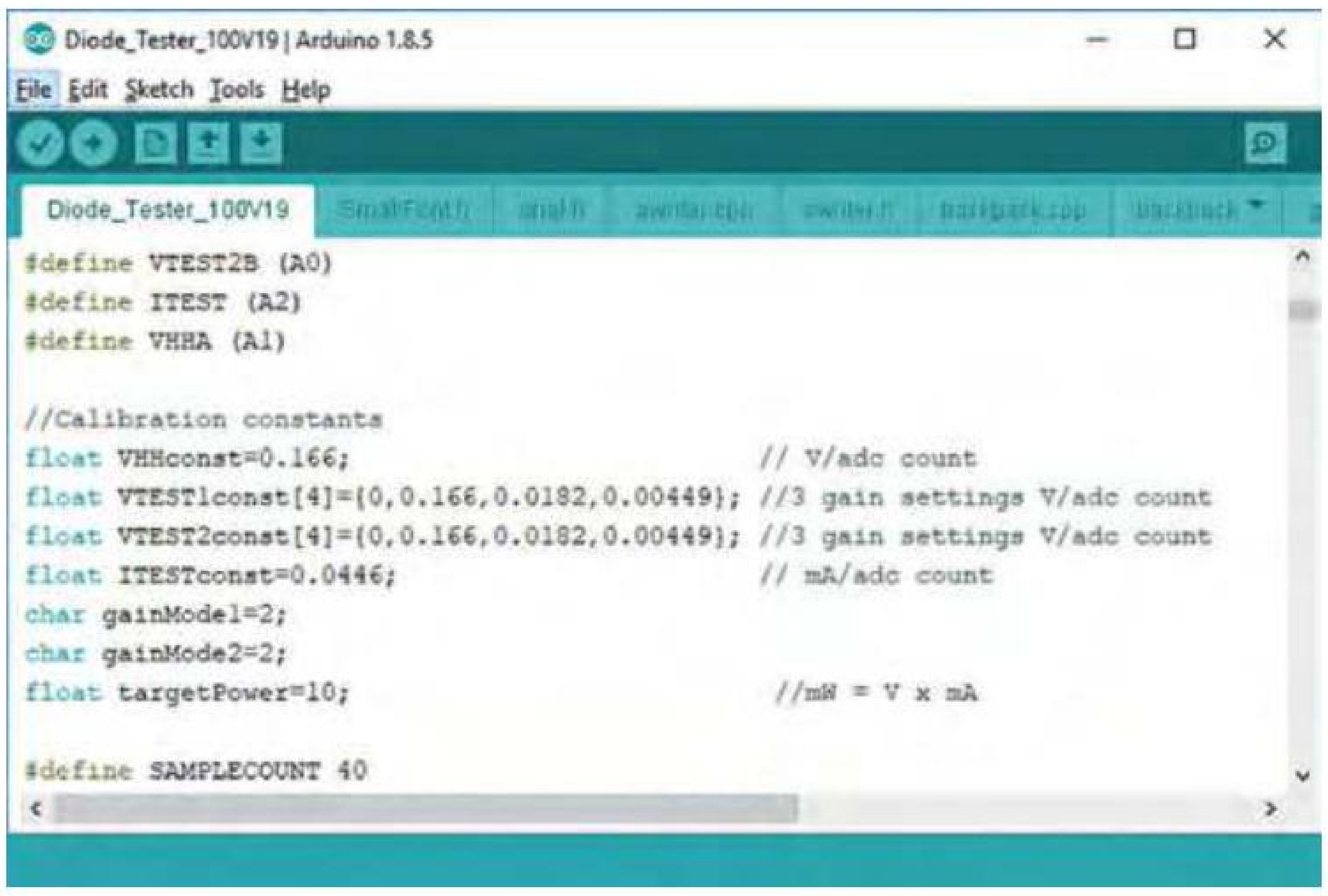
```
      2 for test
      3 for default values
Press anything on serial to exit.

// Use the following defines:
#define TOUCH_X0 1961
#define TOUCH_X1 91
#define TOUCH_Y0 68
#define TOUCH_Y1 2119

Press 1 for calibration
      2 for test
      3 for default values
```

Fig.3: after running the touch calibration sketch and following the instructions, the highlighted text appears on the serial console.


Diode_Tester_100V19 - backpack.h | Arduino 1.8.5
File Edit Sketch Tools Help
backpack.h

```
#define TOUCH_Y1 1896
*/

//touch cals probably need full 'matrix' solution
#define Z_TOUCH_THRESHOLD 100
#define TOUCH_X0 2085
#define TOUCH_X1 145
#define TOUCH_Y0 95
#define TOUCH_Y1 2235

#define TOUCH_OVERSAMPLE 16
```

Fig.4: you then replace this portion of the main sketch with the text copied from Fig.3 above so that it uses the new touchscreen calibration.


Diode_Tester_100V19 | Arduino 1.8.5
File Edit Sketch Tools Help
Diode_Tester_100V19

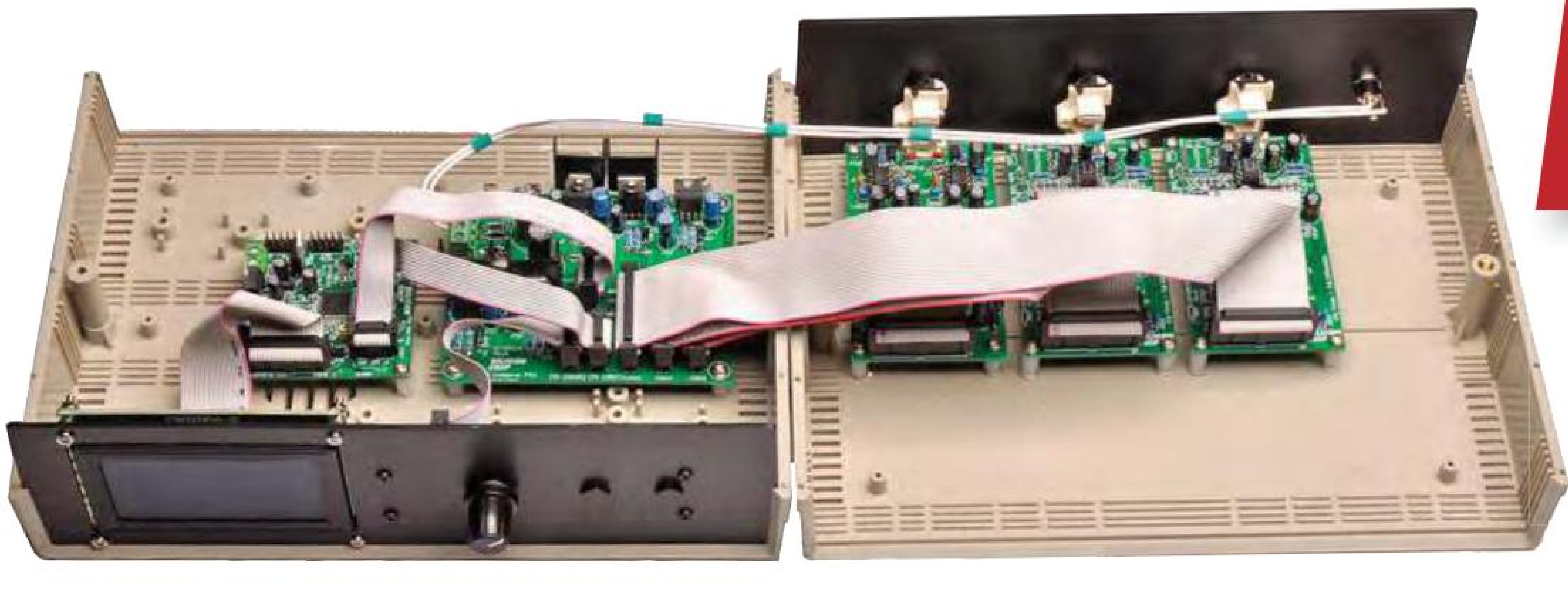
```
#define VTEST2B (A0)
#define ITEST (A2)
#define VHHA (A1)

//Calibration constants
float VHHconst=0.166;                              // V/adc count
float VTEST1const[4]={0,0.166,0.0182,0.00449}; //3 gain settings V/adc count
float VTEST2const[4]={0,0.166,0.0182,0.00449}; //3 gain settings V/adc count
float ITESTconst=0.0446;                           // mA/adc count
char gainMode1=2;
char gainMode2=2;
float targetPower=10;                               //mW = V x mA

#define SAMPLECOUNT 40
```

Fig.5: this is the section of the main sketch where you can change the calibration parameters, just below the comment reading '//Calibration constants'

Any time the 'Back' button is pressed from the Settings page, the settings are saved to EEPROM. The program uses a clever update method so that EEPROM is not rewritten unless necessary, so going into the Settings menu and then exiting without making any changes will not cause any wear on the EEPROM.

In any case, the EEPROM is rated for at least one million rewrites per cell, so you would have to spend a very long time making changes before you're likely to run into any problems with the EEPROM!

Reproduced by arrangement with
SILICON CHIP magazine 2020.
www.siliconchip.com.au



Audio DSP

Part 3

Design Phil Prosser

Words Nicholas Vinen

This is an exciting project, but what exactly is it? A digital signal processor? – a two-way active crossover? – or perhaps an 8-channel parametric equaliser? In fact, it is all of these… and more.

In this third article, we test the *DSP Crossover* modules, then finally connect them together and power the whole unit up. Once it has been tested and assembled in its case, you can set it up before hooking it up between your preamplifier and power amplifier(s), so that it can process the sound.

We explained in the two previous articles that this *DSP Active Crossover* is built from six different modules: a power supply/signal routing module, a CPU board, an analogue-to-digital converter (ADC) board, two identical digital-to-analogue converter (DAC) boards, a front panel control board and a graphical LCD with a small adaptor so that it can connect directly to the CPU board.

Those previous articles described how the circuits of each module worked and how they join together. We also gave the assembly instructions for all the aforementioned modules.

So, if you've been reading along and working as you go, at this stage you should have a complete set of modules – but, you won't have connected any of them together or applied power yet.

Now we get to the fun part: powering everything up, plugging the modules together, and seeing if it works (fingers crossed!).

Once we've verified that everything is working, we can mount all the modules in a case and then we'll explain how to use the resulting device and what sort of performance you can expect from it.

Testing

The first thing to check is that the power supply board is working properly. Regardless of whether you are planning to power the final unit using a plugpack or mains transformer, the easiest way to test it is by wiring a 12V AC plugpack to CON13 on the power supply board, either between pins 1 and 2, or pins 3 and 4.

Don't plug anything else into this board for now.

If you don't have such a plugpack, mount the mains transformer, mains input socket and fuseholder in a metal case (it's usually easiest to place these all in one corner).

Complete and insulate all mains wiring before powering it up, and ensure that the metal case is earthed directly back to the mains input socket or cord.

If using a captive mains power cord, ensure it is adequately clamped to the case using a cord grip grommet or P-clamp, so that pulling on the cord won't allow any internal conductors to come loose.

Fit a fuse with a rating as recommended for the transformer you are using. This may be around 1A, or possibly slightly more if using a toroidal transformer, because these can have a higher inrush current when power is first applied.

During the following testing steps, if you are using a mains power supply, ensure that you can't come into contact with any of the mains conductors while probing the board.

Set your multimeter to a low DC volts range (eg, 20V). Before applying power, check the markings on the board to see where you will be probing. The right-hand end of the 0Ω resistor/wire link below D26 is a convenient place to connect your black ground probe.

You will be checking the voltages at the +9V, –9V, +5V, +3.3V, and VA (5V) pads, as indicated in Fig.11 on page 21 last month, and the PCB itself. These voltages can vary slightly from those indicated.

The acceptable ranges are: 9.2-10.4V (±9V), 4.7-5.4V (+5V, VA) and 3.15-3.6V (+3.3V).

There is no visible indication when the power supply board is powered up. As soon as you have applied power, check the DC voltages at each of the above points.

If any of these are wildly off, check the AC voltage(s) being applied to CON13 and ensure that they are not too far from the nominal 12V. Since the transformer is lightly loaded at this time, readings of 13-14V would not be surprising.

Note that because of the resistor values used to set the regulator output voltages, and because there is no current being drawn from the power supply yet, it is possible that the regulated rails may be even higher than the ranges above suggest.

That's because the worst-case minimum load requirements of the regulators are not catered for with the other boards unplugged.

So if any of the expected readings are below the ranges specified, or well above them, then you should switch off and check for faults.

But if they are slightly too high, you can try connecting a 100Ω resistor from the test point to ground to see if that brings the reading back down into the expected range. If it does, then you can proceed. Otherwise, start looking for soldering or component faults.

Programming the micro

Once you're confident that the power supply is working, if your micro is not already programmed, now is a good

Volume
-28.5dB

An alternative to mounting the unit in the plastic case, as seen opposite, is to use a 19-inch rack mounting case – here seen with a brushed aluminium front panel for a really professional appearance.

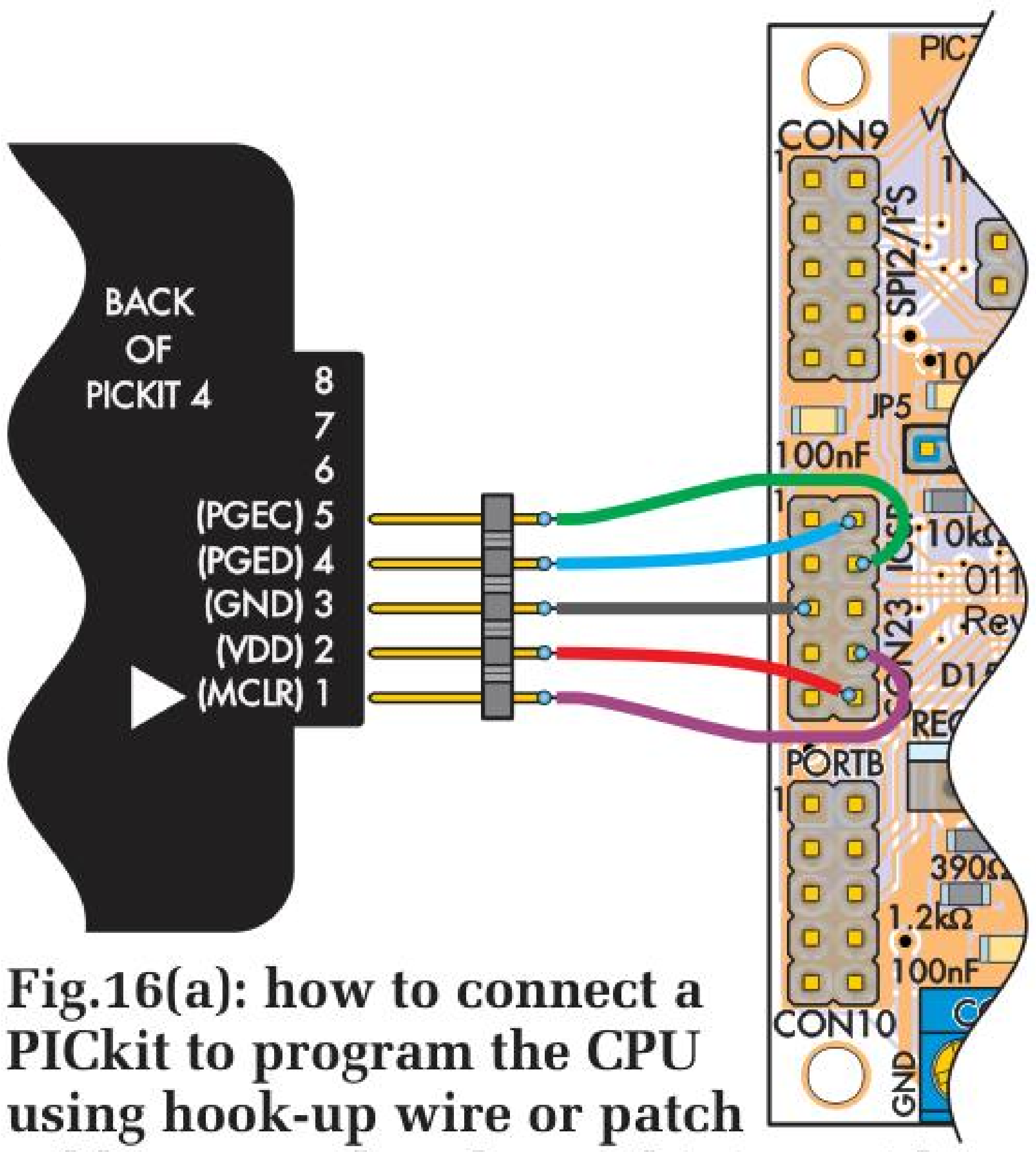

BACK
OF
PICKIT 4
8
7
6
(PGEC) 5
(PGED) 4
(GND) 3
(VDD) 2
(MCLR) 1
CON9
SPI2/I²S
JP5
100nF
10kΩ
CON23
PORTB
390Ω
1.2kΩ
100nF
CON10
GND

Fig.16(a): how to connect a PICkit to program the CPU using hook-up wire or patch cables. Note that the PICkit is upside-down so that pin 1 is at the bottom. Keep the wires short, or programming may fail.

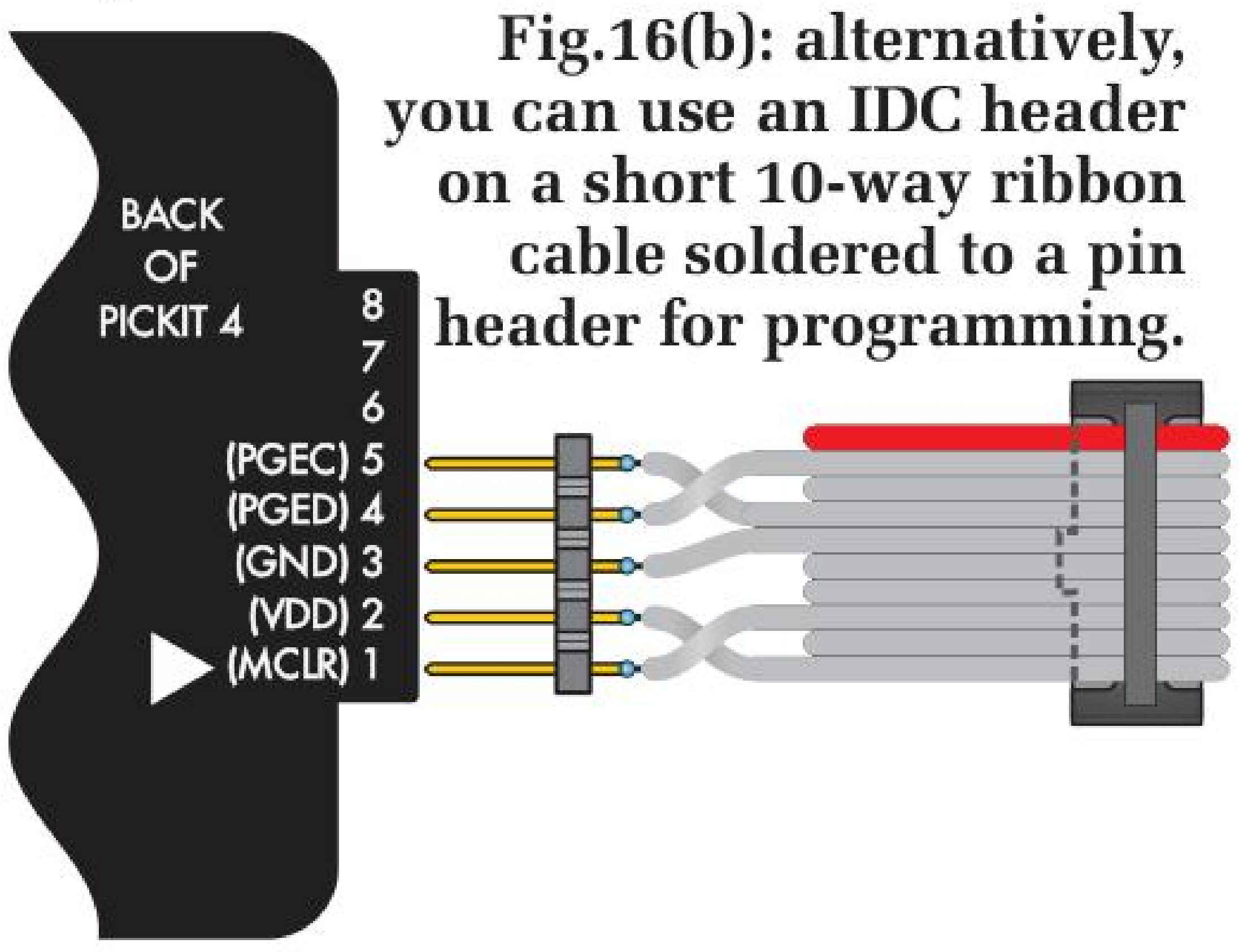

BACK
OF
PICKIT 4
8
7
6
(PGEC) 5
(PGED) 4
(GND) 3
(VDD) 2
(MCLR) 1

Fig.16(b): alternatively, you can use an IDC header on a short 10-way ribbon cable soldered to a pin header for programming.

time to do that. If you have a PICkit 3 or PICkit 4 (or similar), you don’t necessarily need to power the board up to do this; the programmer can supply power to program the chip, and indeed, it is safer to do it this way.

As mentioned last month, the programming header (CON23) does not have the same pinout as the PICkit 3/4, so you need to make up an adaptor to connect it. This could be as simple as five male/female jumper leads plugged into CON23 at one end, and the appropriate PICkit pin at the other end.

Or, you could crimp a 10-pin IDC line socket onto a spare section of 10-way ribbon cable, then separate the wires at the other end, cut some off short and solder the others to a 5-pin header. You can then plug the PICkit into that header.

To program the chip in our prototype, we soldered a 5×2-pin box header onto a small piece of veroboard, along with a 5-pin right-angle header, and then made the five required connections using short lengths of Kynar (wire-wrap wire) soldered between the pads.

Regardless of the method you choose, the required cable configuration is shown in Fig.16(a) and Fig.16(b). Remove jumper JP5 during programming and re-insert it when finished.

If using a PICkit, you can load the HEX file into the PIC32MZ chip using the free Microchip MPLAB IPE software, which is installed along with the MPLAB IDE (also a free download). Grab this from the following link: **microchip.com/mplab/mplab-x-ide**

Having installed the IPE (if you don’t have it already), launch it and change the Device field to ‘PIC32MZ2048EFH064’ (see Fig.17). If you can’t find that device in the list, you need to update to the latest version of the software.

Plug in your programming tool, then select it from the list and click ‘Apply’, then ‘Connect’. If your tool does not support this chip, you will get a message saying so.

You may get an error message saying that no power was detected and the connection has failed. This is fine, as we want to ensure that the PICkit is set up correctly before applying power to the chip.

Now, to the right of ‘Hex File’: below, click ‘Browse’ and select the HEX file which you have unzipped from the download package for this project, obtained from the March 2020 page of the *PE* website.

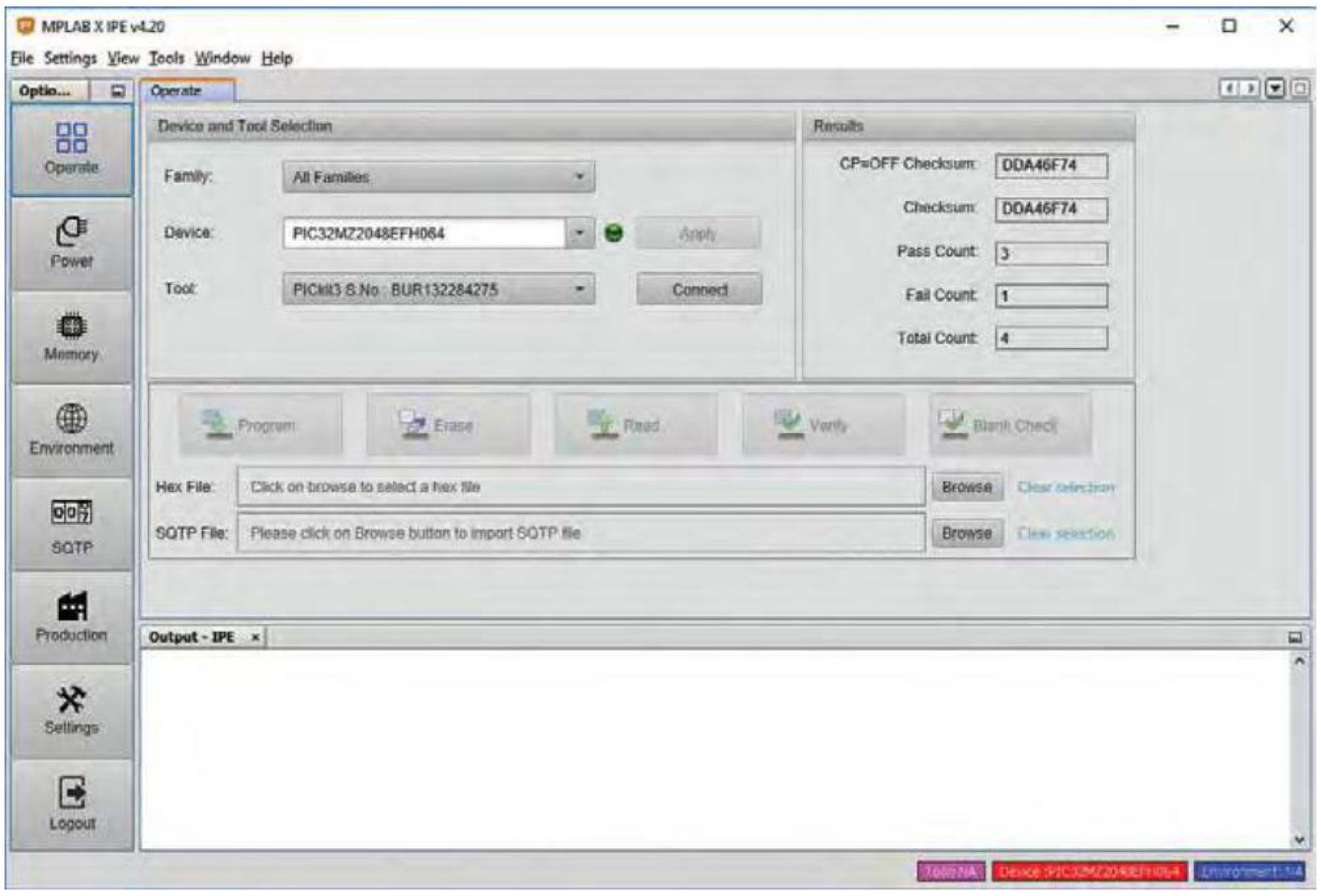
Next, click on the ‘Power’ tab on the left side of the screen. You may need to switch the software to ‘Advanced Mode’ to access this tab. Ensure that the ‘Power Target circuit from Tool’ option is ticked (Fig.18).

Switch back to the ‘Operate’ tab, check that your programmer is connected to the CPU board correctly (if not, click the ‘Connect’ button again) and press the ‘Program’ button. You will get a series of messages at the bottom of the screen indicating the progress (Fig.19).

If programming failed or you get a message that the software is unable to detect or connect to the target device, check your wiring. If that’s good, then you may have a problem with the soldering of IC11 or some associated components, or you may have one or more solder bridges on the board. Examine it carefully for faults.

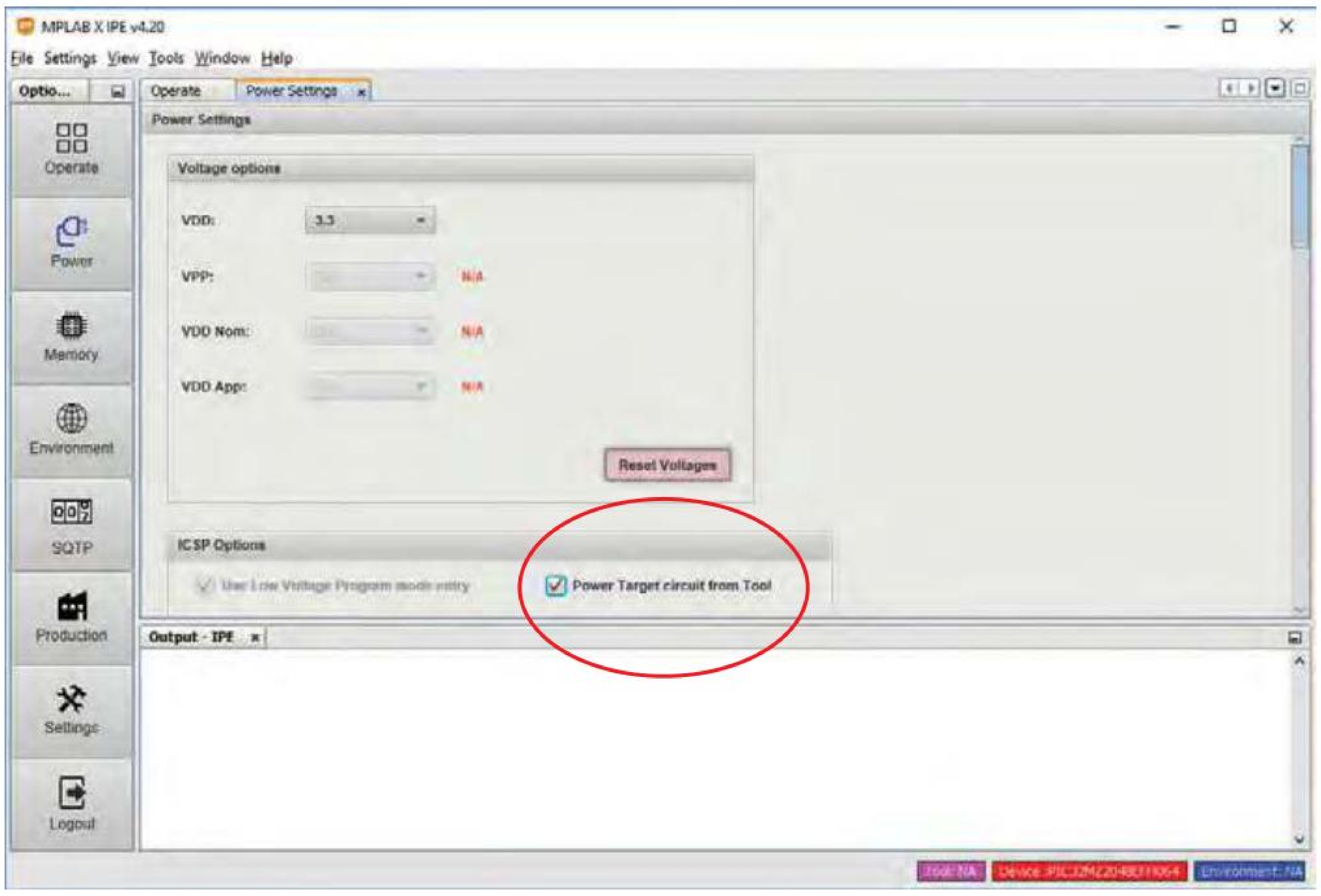
MICROCHIP
MPLAB
PICKIT 4
IN-CIRCUIT DEBUGGER

Programming with a PICkit 4 is much faster than with a PICkit 3, which is especially helpful in this project, as the HEX file is rather large – 2MB.


MPLAB X IPE v4.20
File
Settings
View
Tools
Window
Help
Optio...
Operate
Operate
Power
Memory
Environment
SQTP
Production
Settings
Logout
Device and Tool Selection
Results
Apply
Connect
Program
Erase
Read
Verify
Blank Check
Browse
Clear selection
Browse
Clear selection
Output - IPE

Family: All Families
Device: PIC32MZ2048EFH064
Tool: PICkit3 S.No : BUR132284275
CP=OFF Checksum: DDA46F74
Checksum: DDA46F74
Pass Count: 3
Fail Count: 1
Total Count: 4
Hex File: Click on browse to select a hex file
SQTP File: Please click on Browse button to import SQTP file

Fig.17: the first step to set up MPLAB X IPE is to select the correct PIC chip, as shown here, and check that it has detected your programmer.



Power Settings

Voltage options

VDD: 3.3

VPP: N/A

VDD Nom: N/A

VDD App: N/A

ICSP Options

☑ Use Low Voltage Program mode entry

☑ Power Target circuit from Tool

Fig.18: to make things easier, rather than power the board externally, the PICkit can supply power to IC11 during programming, as long as you have checked this box.

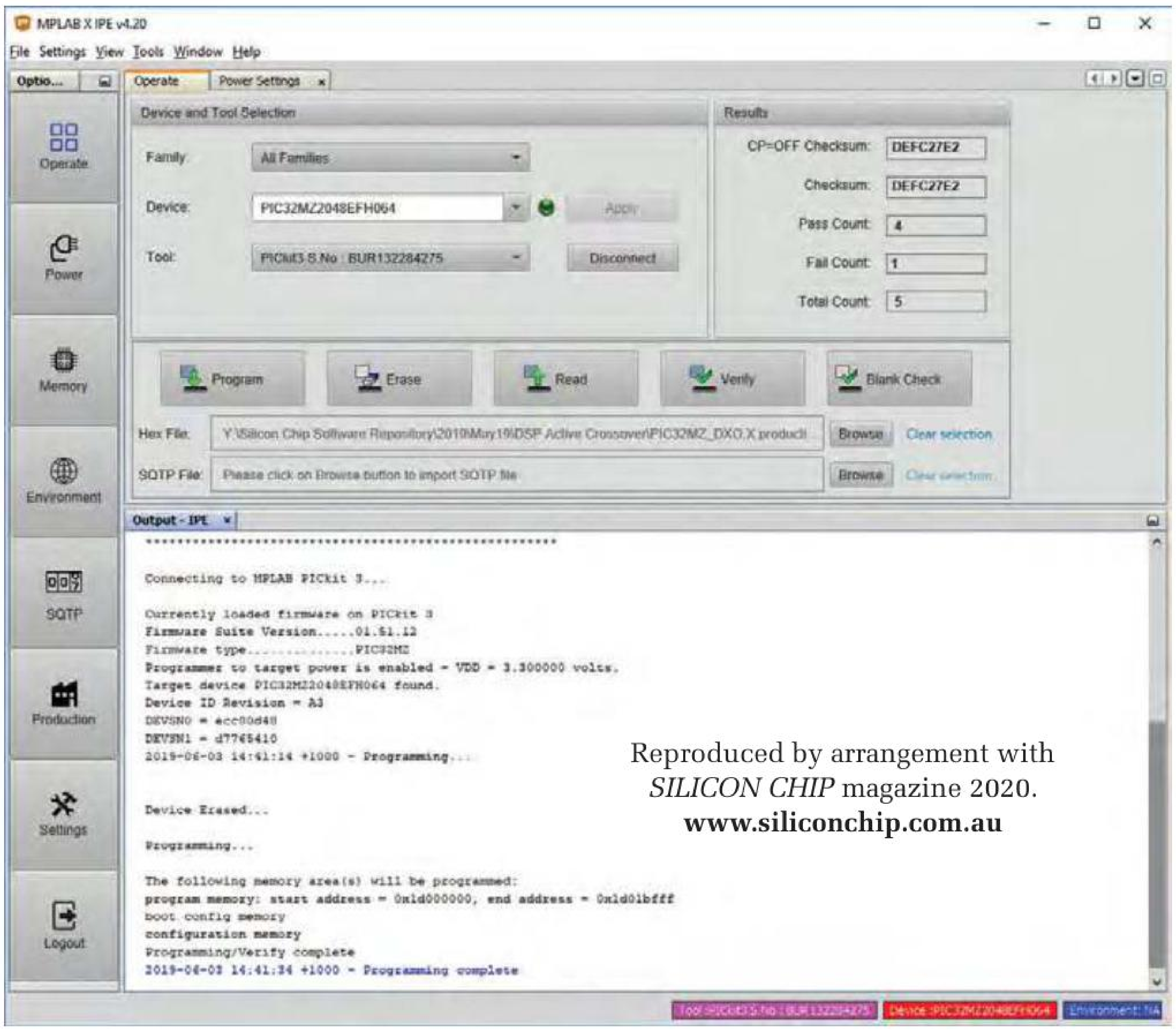
Our first attempt to program the chip in the prototype failed. We examined all the pins of IC11 under magnification, but couldn't see any obvious problems like bridges or unsoldered pins. We solved this by adding flux paste to all the pins of IC11 and then re-flowing the solder using a hot-air rework station. So that is worth trying if you can't figure out why it isn't working.

We suspect that the solder on one of the pins on our chip hadn't flowed down onto the pad below, but it's hard to say for sure. Whatever the problem was, it's gone now.

Assuming IC11 is soldered correctly, and your programmer is wired up as shown, the chip should be successfully programmed and verified. You can then move on to the next stage of testing.

Further testing

The next step is to test the control circuitry. You will now need the three 10-wire ribbon cables you made up earlier (described at the end of last month's article). In each case, make sure that the pin 1 triangle/red wire goes to pin 1 on the connector that you're plugging it into, and note that it's possible to plug in the IDC headers offset, so that some of the pins are not connected – you must avoid doing that.



Device and Tool Selection

Family: All Families

Device: PIC32MZ2048EFH064

Tool: PICkit3 S.No : BUR132284275

Results

CP=OFF Checksum: DEFC27E2

Checksum: DEFC27E2

Pass Count: 4

Fail Count: 1

Total Count: 5

Hex File: Y:\Silicon Chip Software Repository\2019\May19\DSP Active Crossover\PIC32MZ_DXO.X producti

SQTP File: Please click on Browse button to import SQTP file

```
*****************************************************

Connecting to MPLAB PICkit 3...

Currently loaded firmware on PICkit 3
Firmware Suite Version.....01.51.12
Firmware type..............PIC32MZ
Programmer to target power is enabled - VDD = 3.300000 volts.
Target device PIC32MZ2048EFH064 found.
Device ID Revision = A3
DEVSN0 = acc00d48
DEVSN1 = d7765410
2019-06-03 14:41:14 +1000 - Programming...


Device Erased...

Programming...

The following memory area(s) will be programmed:
program memory: start address = 0x1d000000, end address = 0x1d01bfff
boot config memory
configuration memory
Programming/Verify complete
2019-06-03 14:41:34 +1000 - Programming complete
```

Reproduced by arrangement with
SILICON CHIP magazine 2020.
www.siliconchip.com.au

Fig.19: now we can load our HEX file, connect to the PIC and program it. If successful, you should get the same messages in the bottom pane as we did here.

The two shorter cables connect from CON7 on the power supply board to CON17 on the CPU board, and from CON18 on the power supply board to CON11 on the CPU board. On the power supply board, pin 1 of each connector is at bottom right. On the CPU board, pin 1 of CON7 is near D16, while pin 1 of CON11 is near the 10µF capacitor.

The third, longer cable connects from CON19 on the power supply board to CON20 on the back of the front panel interface board. Again, make sure that the pin 1s are wired correctly. Pin 1 of CON20 is near to rotary encoder RE1.

You will also need to wire up the LCD screen. This is done using the 20-way ribbon cable. Plug one end into CON8 on the CPU board (pin 1 is next to the mounting hole in the lower-right corner of the PCB) and connect the other end to the small LCD adaptor, which you will have already soldered to the back of the screen. Pin 1 is marked on that PCB.

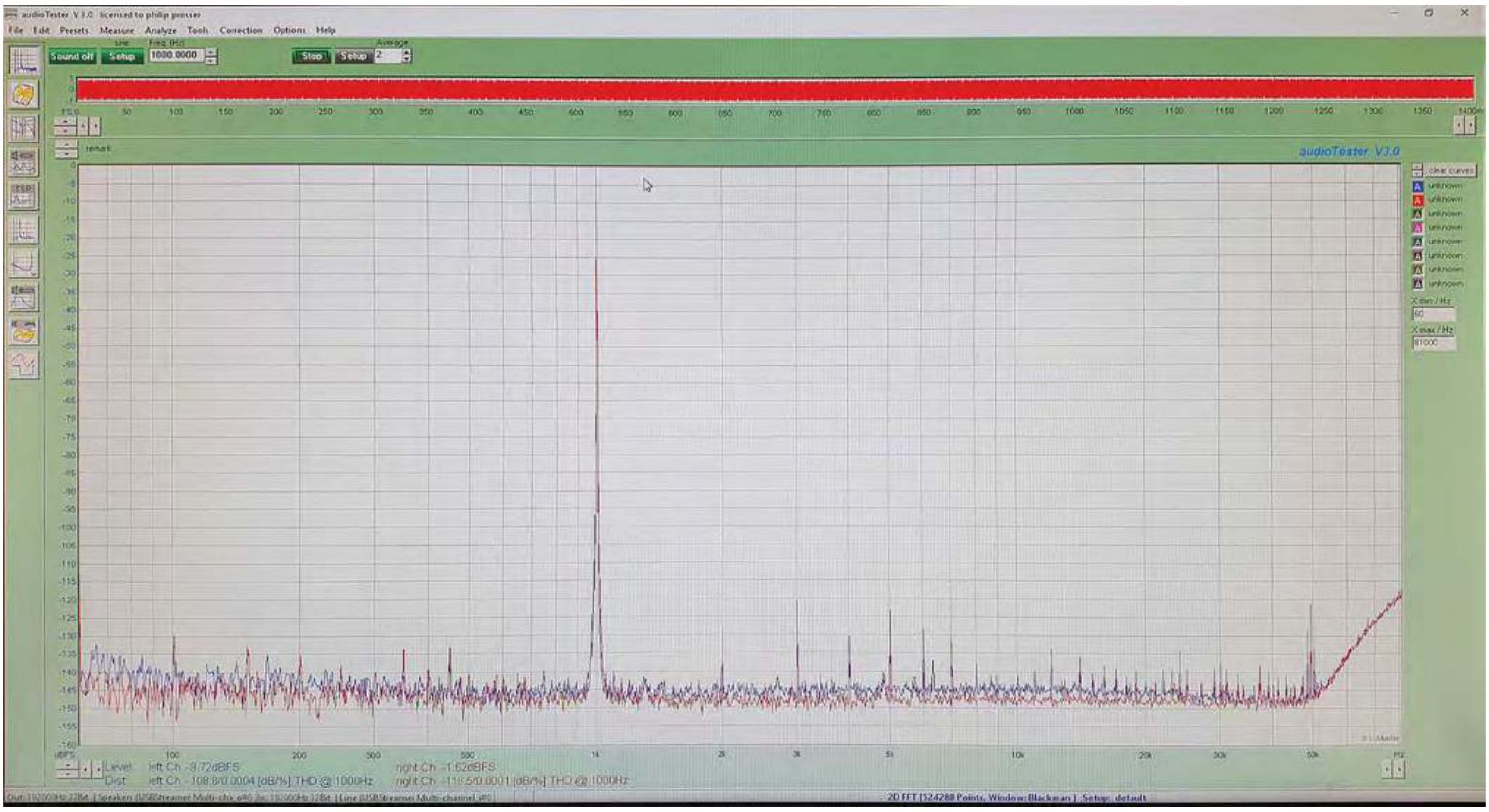
If you don't have that adaptor, you can separate the wires in the ribbon cable and solder them to the 20 pins on the LCD screen module, with the red wire to pin 1 and so on. That's how the original prototype was built, but it's a tedious process, hence the adaptor board.

You can now apply power and check that the LCD screen lights up and you get a sensible display on the screen. You will need to adjust contrast trimpot VR1 before you see anything on the screen. Also check that LK2 is in the V_{EE} position. Turn the rotary encoder and check that you can scroll through the menus, and that pressing the front panel buttons gives the expected results.

A lack of display on the screen could be due to several problems. If you programmed the microcontroller yourself, you know that it is at least running, but there could be a soldering fault on one of the pins connecting to the LCD, or there could be a wiring problem with the cable.

LED2 on the CPU board should flicker when the display is updated, and you can force this to happen by turning the rotary encoder knob.

As the CPU board has two onboard regulators and generates its own 3.3V rail, if it doesn't work straight away, then it's a good idea to check that first. The left-hand pin of CON5, labelled GND on the PCB, makes a good reference point. There is a via between


audioTester V3.0 licensed to philip prosser
File Edit Presets Measure Analyze Tools Correction Options Help
Sound off
Setup
Stop
Setup
audioTester V3.0
clear curves
unknown
dBFS
100
200
300
500
1k
2k
3k
5k
10k
20k
30k
50k
Hz
2D FFT [524288 Points, Window: Blackman] ; Setup: default

Level left Ch -9.72dBFS right Ch -1.52dBFS
Dist left Ch -108.8/0.0004 [dB/%] THD @ 1000Hz right Ch -118.5/0.0001 [dB/%] THD @ 1000Hz

Fig.20: a PC-based spectrum analyser plot showing the output of the *DSP Active Crossove*r when fed with a (near) pure sinewave. THD readings are shown at bottom; note that these were not done with a full-scale signal (which likely would give better results) but also, they do not incorporate noise (ie, they are not THD+N readings).

CON5 (near the GND terminal) and CON10 which connects to the +5V rail from the power supply, so check this voltage first.

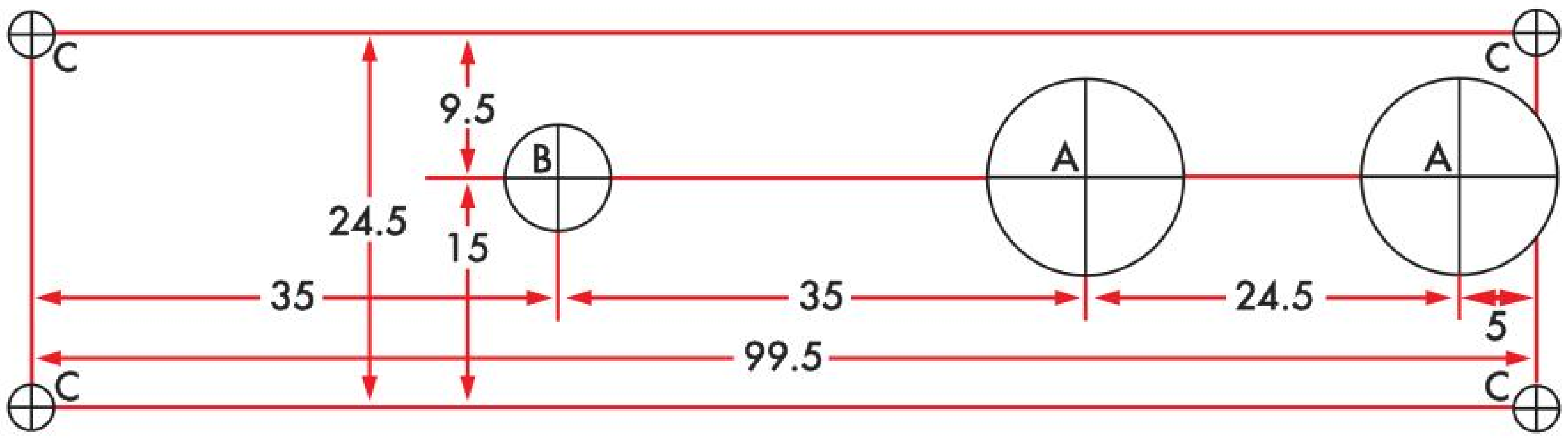
Next, check the voltage on the other terminal of CON5. You should get a slightly lower reading, of around 4.7-4.8V, due to the forward voltage drop of D15. Next, to check the 3.3V rail, probe either of the vias immediately to the left of the PIC, IC11. The easiest one to reach is the one just to the right of the capacitors to the right of JP5.

Expect a reading of 3.17-3.58V. Anything outside this range suggests a problem with regulator REG2 or one of its associated components. Switch off and check the board carefully.

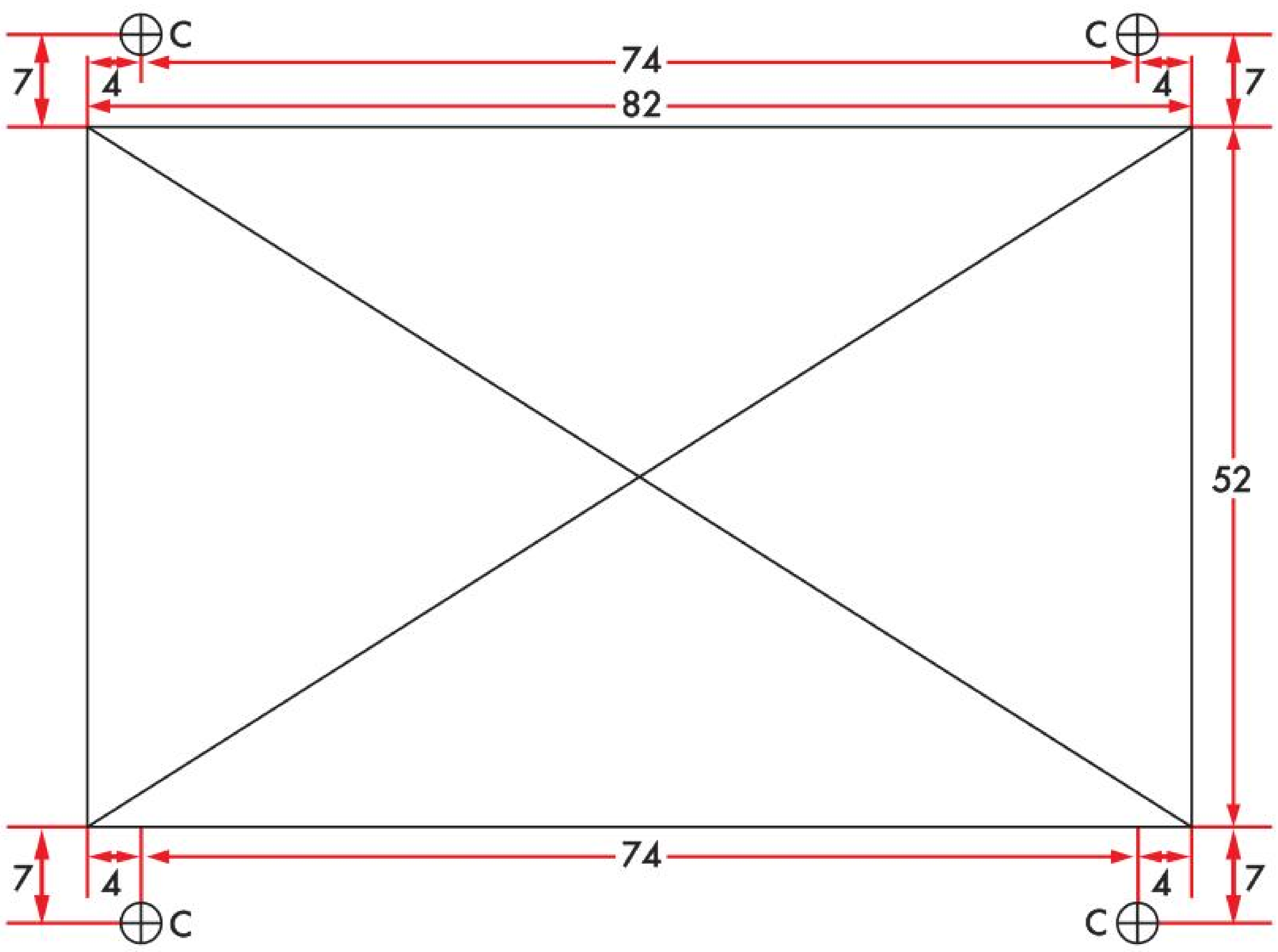
If the power supply rails check out, it's a good idea to verify that the primary oscillator is running. You will need a frequency meter which goes up to at least 8MHz; some DMMs have this function. Using the same ground point as a reference, probe the left-hand end of the 470Ω resistor near the bottom right-hand corner of IC11. You should get a reading close to 8MHz.

If you don't, then either IC11's oscillator amplifier is not operating (suggesting a problem with the chip, its soldering or its programming) or there is a problem with crystal X2.

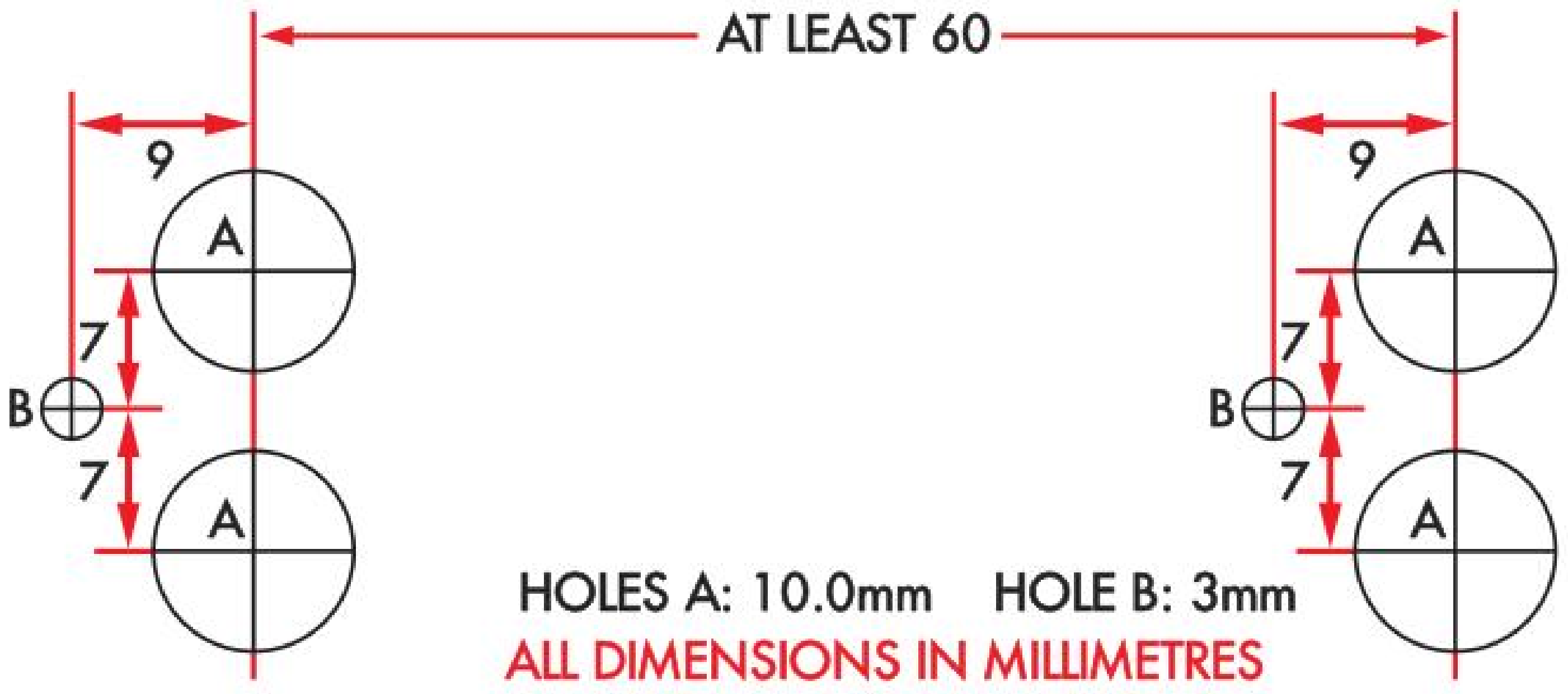
If you are seeing the 8MHz signal but still not getting anything on the LCD, that suggests a connection problem between the chip and the LCD, so check all the headers and cables. If LED2 is not flickering, IC11 may not be programmed correctly or there is a bad connection somewhere, probably on the CPU board.


C
C
C
C
B
A
A
9.5
24.5
15
35
35
24.5
5
99.5

HOLES A: 13.0mm HOLE B: 7mm HOLES C: 3mm ALL DIMENSIONS IN MILLIMETRES

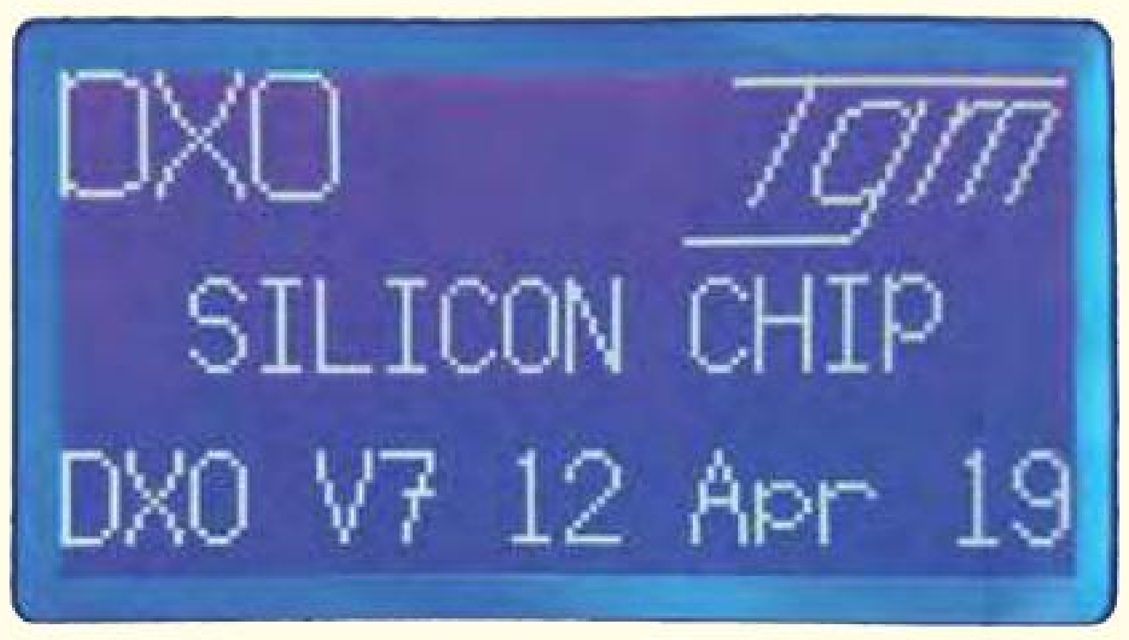

C
C
C
C
7
4
74
82
4
7
52
7
4
74
4
7

Above left (Fig.21) are the three holes required in the front panel controls, which are all mounted on the front panel PCB – the two pushbutton switches (S1; 'Exit' and S2; 'Select') and the Rotary Encoder. Exact positioning on the panel is unimportant as the front panel PCB determines the position. At bottom left (Fig.22) is the cutout for the liquid crystal display, while below (Fig.23) are the four holes required for two pairs of RCA sockets (the third set would be identical, but the separation may vary).

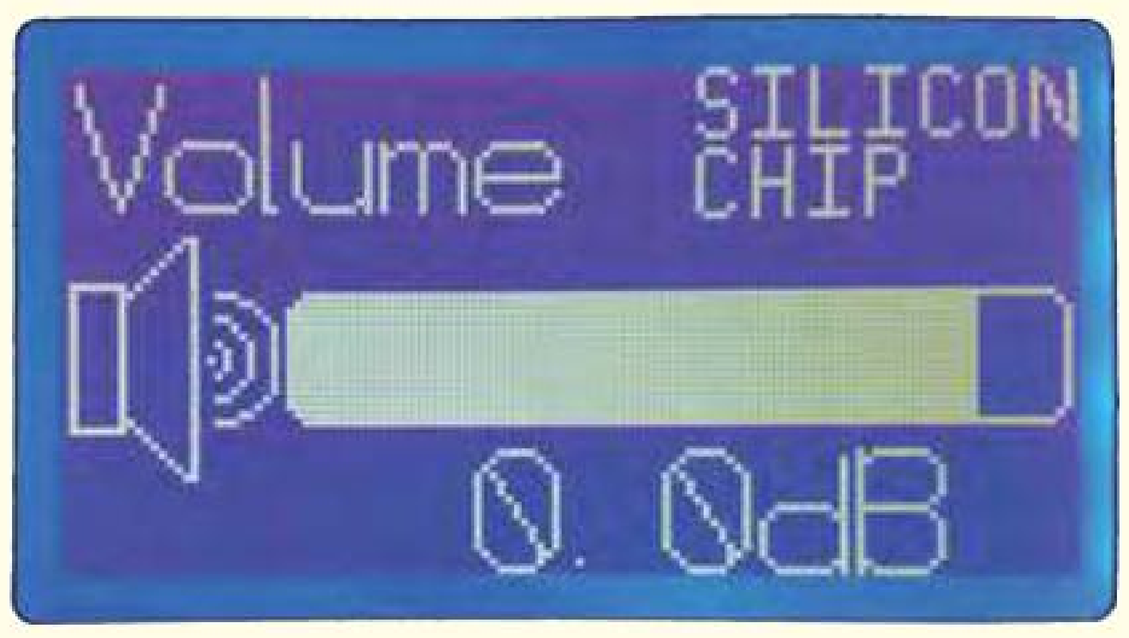

AT LEAST 60
9
A
7
B
7
A
9
A
7
B
7
A
HOLES A: 10.0mm HOLE B: 3mm
ALL DIMENSIONS IN MILLIMETRES

SILICON
CHIP

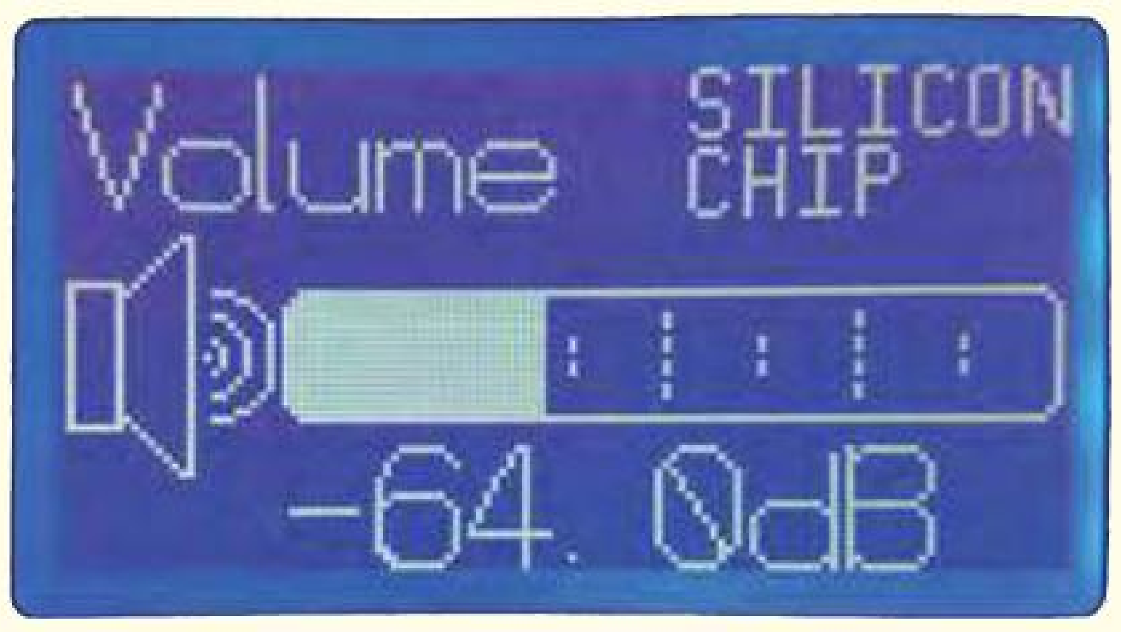
Screen01: the initial splash screen, which is quickly followed by…



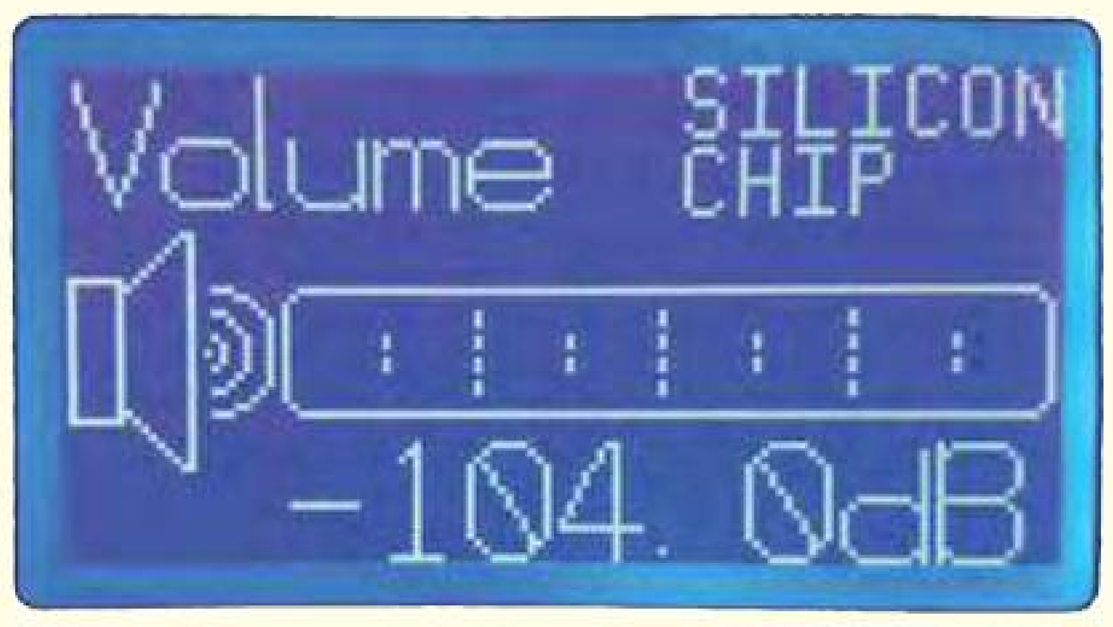
Screen02: a second splash screen, giving the software version and build date, which is then followed by…



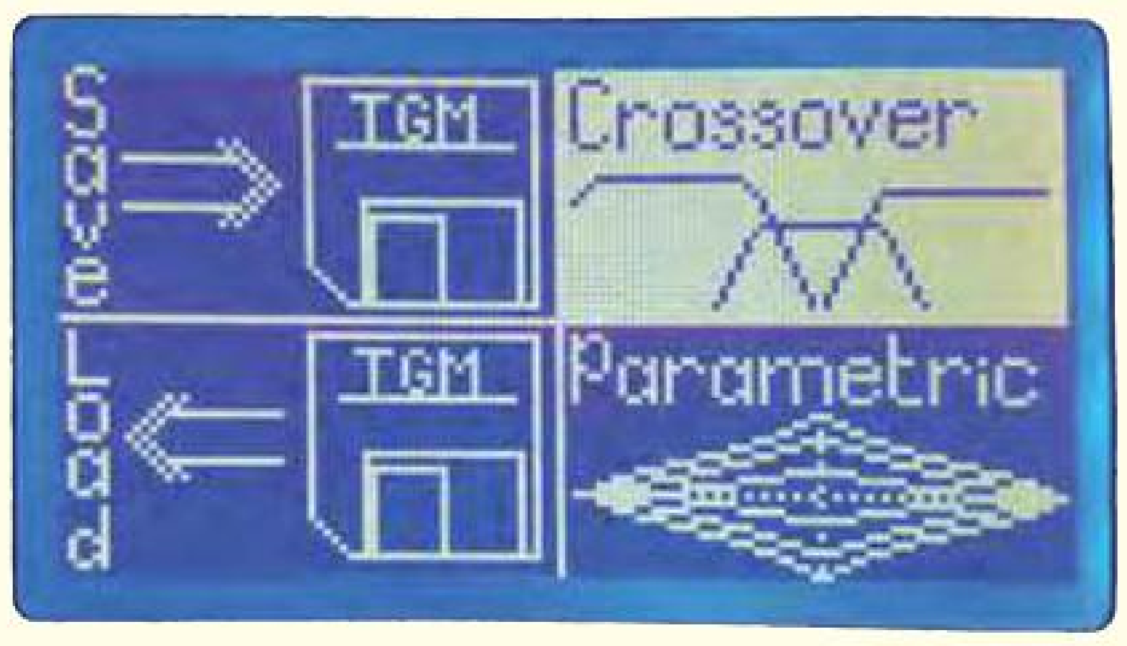
Screen03: the default screen, which gives volume control and starts at 0dB. Rotate the encoder knob to...



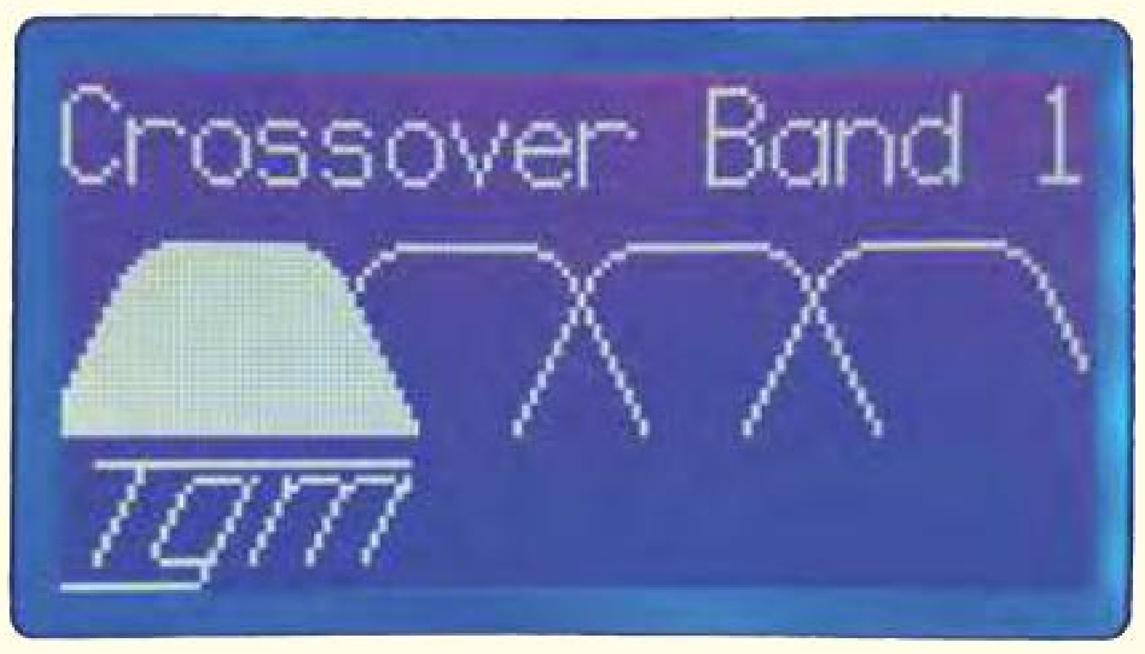
Screen04: adjust the volume. It can go up as high as +12dB or down as low as…



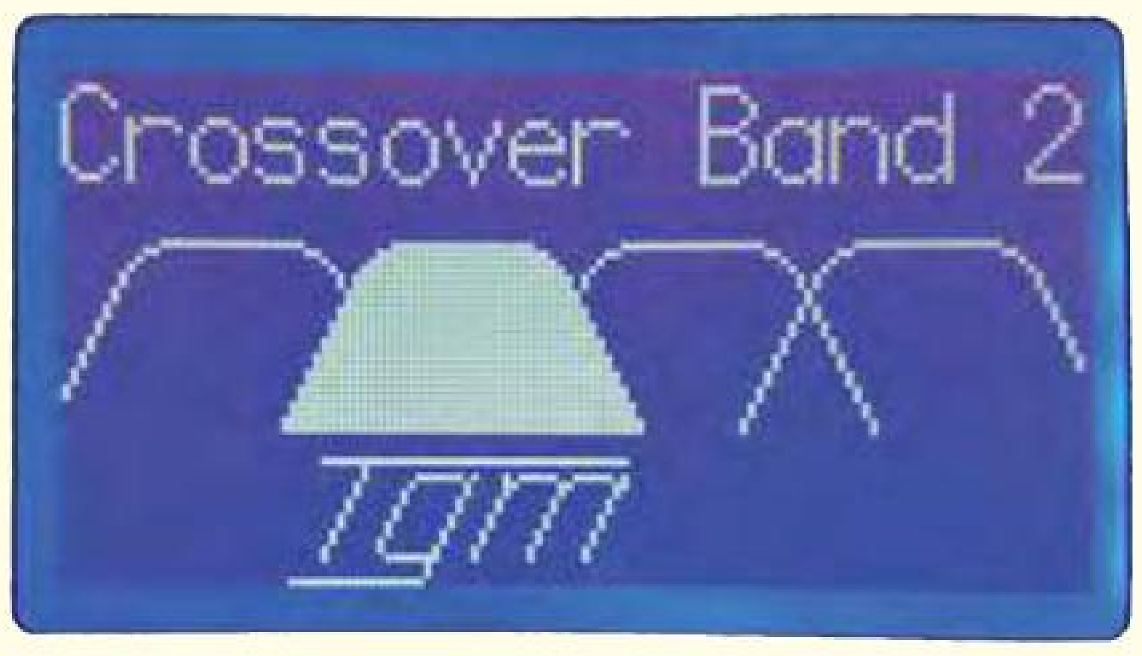
Screen05: –104dB. Pressing either pushbutton (or the knob) on this screen takes you to…



Screen06: the main menu, which has four options. Use the rotary encoder to change the current selection and press S2 or the knob to go into that sub-menu.



Screen07: in the crossover sub-menu, first you select which band you want to adjust using the rotary encoder (you can still adjust other bands after making the initial selection).



Screen08: here we've selected Band 2. Only two bands are initially available. You need to change other settings to activate Bands 3 and 4. Selecting a band takes you to…

It's also possible that LED2 has been installed backwards. If you've verified its orientation and the chip programming, and it still isn't lighting up, check your soldering carefully.

Plugging the rest of the boards in

Assuming you have had success with the LCD and controls, you can now connect the other three boards. As shown in Fig.6 on page 23 of the January 2020 issue, CON16 connects to CON2 on the ADC board, while CON14 goes to CON3 on the first DAC board (woofer output) and CON15 goes to CON3 on the second DAC board (tweeter output).

As with the other cables, be careful to make sure that the pin 1 side of each plug goes to the pin 1 marked for each header, and that you don't plug them in offset by one row of pins.

All the ADC and DAC boards have pin 1 on the side of the header closest to the nearest edge of the board, and similarly, on the power supply/routing board, pin 1 of each header is towards the bottom edge.

We specified three different cable lengths last month, since these three boards will be different distances from the power supply module.

In our prototypes, the ADC board is closest, so it uses the shortest cable; however, there's nothing to stop you from using a different arrangement.

Once those are all plugged in, check that JP1-JP4 are inserted and that LK1 is set to SDO4. The only way to really test it is to connect a signal source to the ADC inputs, power the unit up and check that you're getting appropriate signals from the four outputs, using either a scope or a power amplifier and speakers.

If you are using an amplifier, turn the volume down initially, in case there's something wrong; otherwise, your ears may get blasted!

If you don't get the expected result, check that all the jumpers are in the correct positions (see last month).

Preparing the rear panel

The steps for final assembly are: drill and cut holes in the front and rear of the case, determine the ideal location for each module and mount them to the case, attach the LCD and control board to the front panel and then complete the wiring.

On the rear panel, you will need to drill six holes of 9-10mm diameter for the RCA sockets. Ideally, you should also drill a 3mm hole for each pair of RCA sockets, to mount the connector to the rear panel so that it isn't damaged when pushing the plugs in. The hole pattern required is shown in Fig.23.

Each group of holes will need to be at least 60mm apart, to give room for the boards to fit side-by-side. You may wish to increase the space between the ADC module and the two DAC modules (assuming your case is large enough), to make the distinction more obvious.

On the rear panel, you will also need to mount either a concentric socket for a plugpack or a mains cord or socket (ie, an IEC input socket).

While it's a good idea to also fit a fuseholder to the rear panel for the plugpack-powered version, it isn't strictly necessary. However, you definitely need a fuse if using a mains power supply. Our second prototype, shown in the photos here, is plugpack-powered.

Mains wiring

For a mains supply, if you're fitting an IEC socket for convenience (wired-in or 'captive' mains cords can be a bit of a pain), you can use one with an integral fuse and then you won't need to mount a separate fuseholder.

But note that IEC sockets with fuse holders often have exposed, live conductors on the inside, so it's a good idea to apply neutral-cure silicone sealant in these areas so that they are not a shock hazard if you operate the device with the case open, during testing.

It is somewhat easier to drill a hole to suit a wired-in mains cable, and that is a valid approach; just make sure you fit a proper 'safety' fuseholder wired in series with the live lead, and that you provide adequate clamping to ensure the mains cord can't be accidentally pulled out, even if the unit is dropped.

The best way to do this is either using a cord grip grommet (although this does require a properly profiled hole to be made) or an appropriately sized cable gland. If using a cable gland, it's best to fit the part which tightens up around the cable on the inside of the case, so it can't be loosened from the outside. Alternatively, apply superglue (cyanoacrylate) to the threads before tightening it up.

Another thing that's necessary if you are using a mains power supply in a metal case is to properly earth the case. Run a short green/yellow striped earth wire (stripped from a section of 13A-rated mains cable) directly from the mains input socket to a chassis-mounting eyelet or spade lug. If the case is painted, scrape the paint away around the lug mounting point.

Use the largest diameter screw possible to attach this lug, along with shakeproof washers and two nuts. If using a captive mains cord, simply separate its earth wire and run it to this chassis earth lug. You do not need to make an earth connection anywhere else in the device.

You also need to ensure that there is good electrical continuity between the various case panels when the case is assembled. This may require removing some paint where the panels are screwed together, or otherwise attached.

Verify that you have a low resistance between any exposed metal on the case and the mains earth pin before powering the unit up.

Mounting the modules

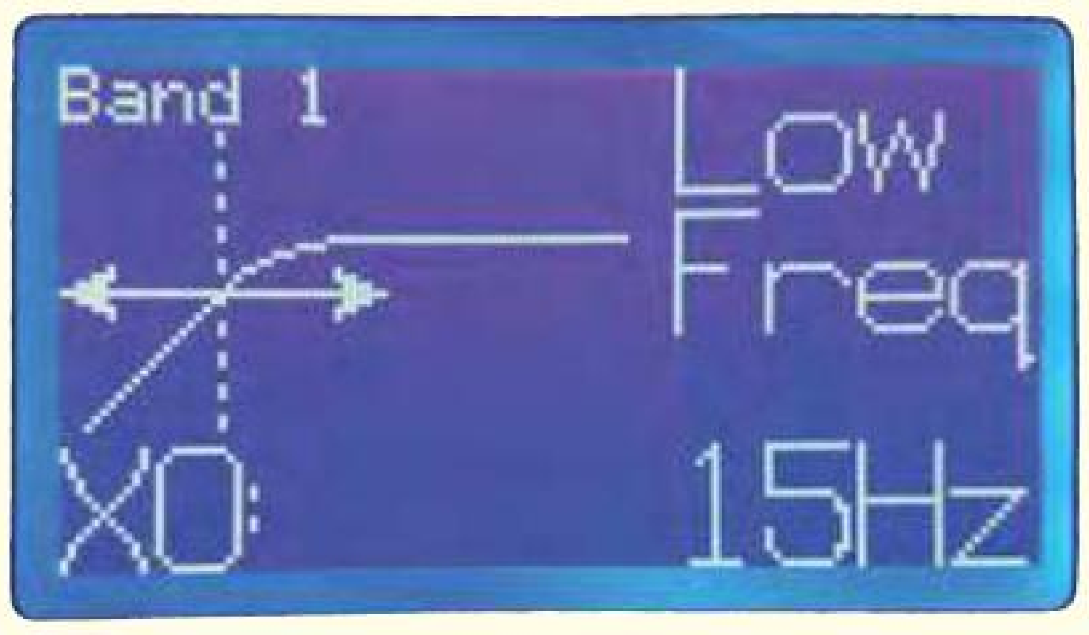
Once you have made the holes in the rear panel and attached and wired up any required power supply components, you can mark out the mounting hole positions for the power supply board, CPU board, ADC board and DAC boards. Drill these to 3mm, deburr, then attach the modules using machine screws and tapped spacers. You can then wire them back up, as you did during the testing.

That just leaves the LCD and front panel control module to mount. You need to make a rectangular cutout 82mm wide and 52mm tall in the front panel for the LCD screen to fit through. (See Fig.22).

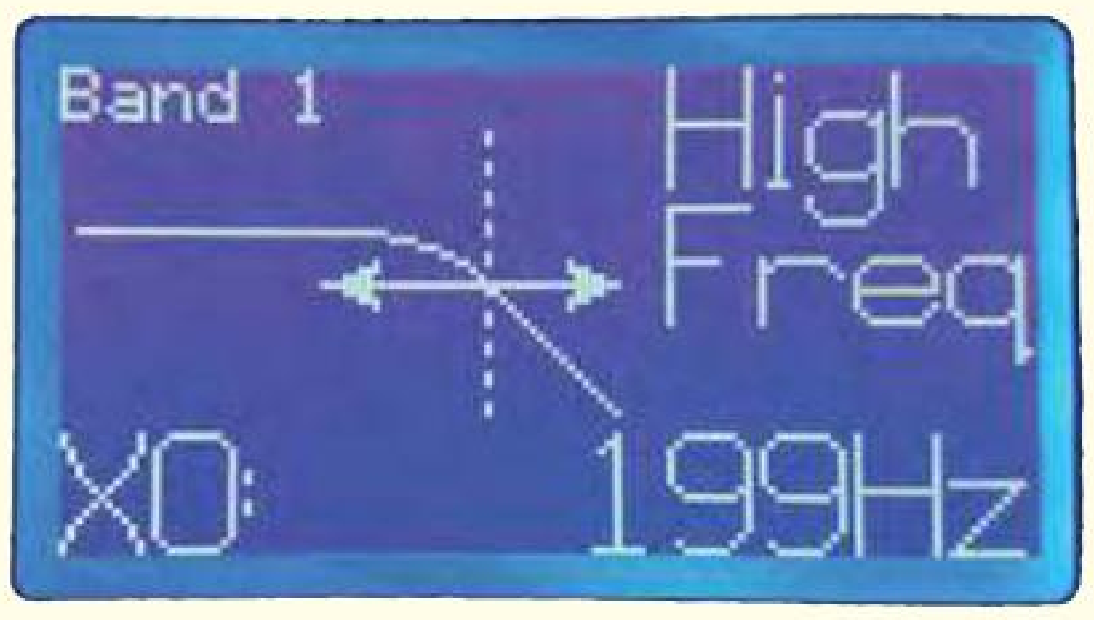
Make sure it's centred vertically on the panel, and at least 5mm from any protrusions on either side, as the LCD board is slightly larger than the screen (92mm × 70mm).

You can draw the required outline on the panel and then cut it out using a rotary cutting tool like a Dremel. Or you could drill a hole and then use a nibbling tool. Either way, file the edges smooth and make sure that the panel fits, then mark out and drill the four 3mm corner mounting holes. You can then attach the panel using 16mm M3 machine screws, nuts and washers.

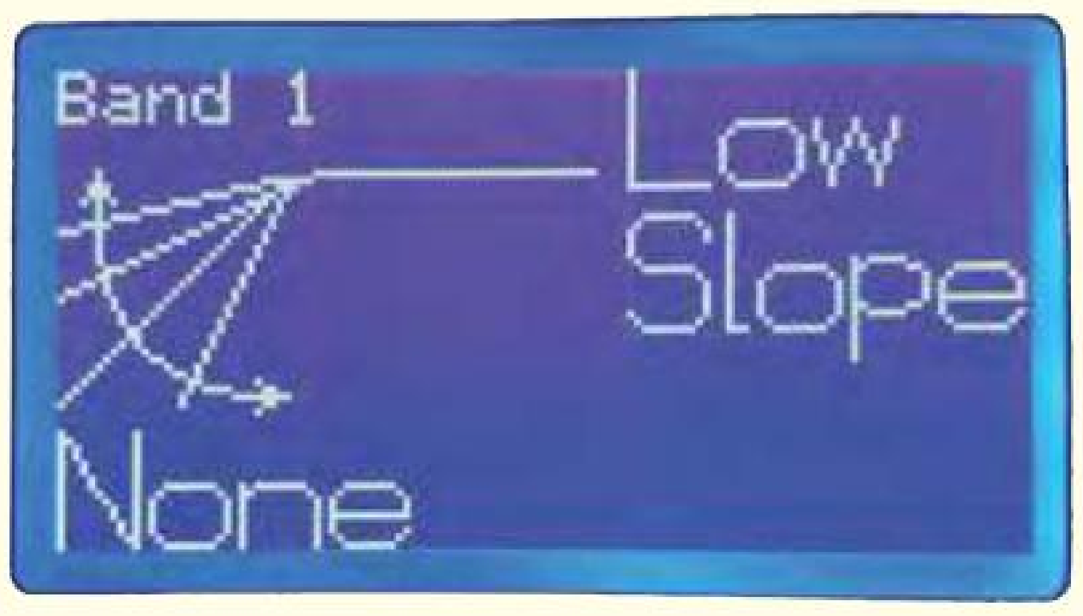
Extra nuts and/or washers can be used to space the LCD board out from the panel (see the photo on the first page of this article).



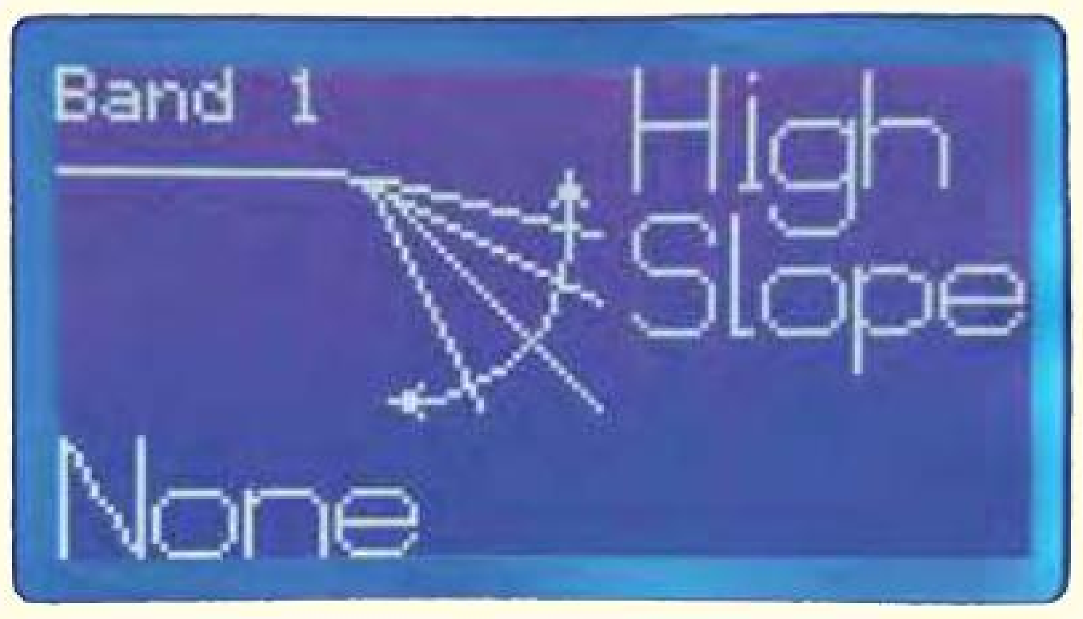
... Screen09: the first adjustment, which allows you to adjust the lower −3dB point using the rotary encoder, to as low as 15Hz. Pressing S1 will take you back to the volume screen, or press S2 to go to...



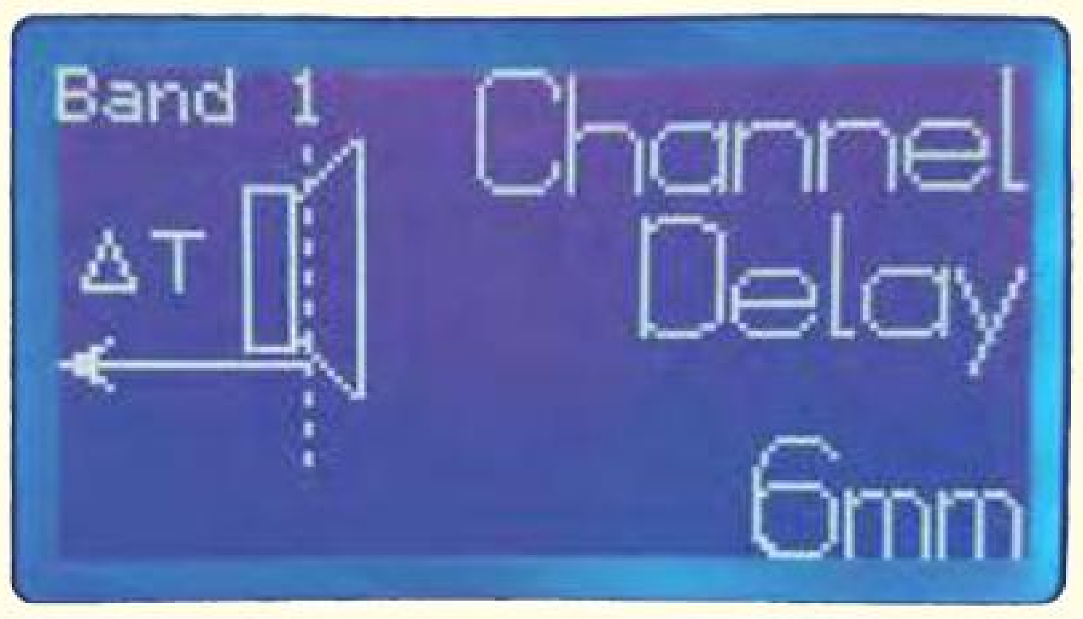
Screen10: the second crossover adjustment, the upper −3dB point, which goes as high as 15kHz. Here it is set to 199Hz. Pressing S2 takes you to…



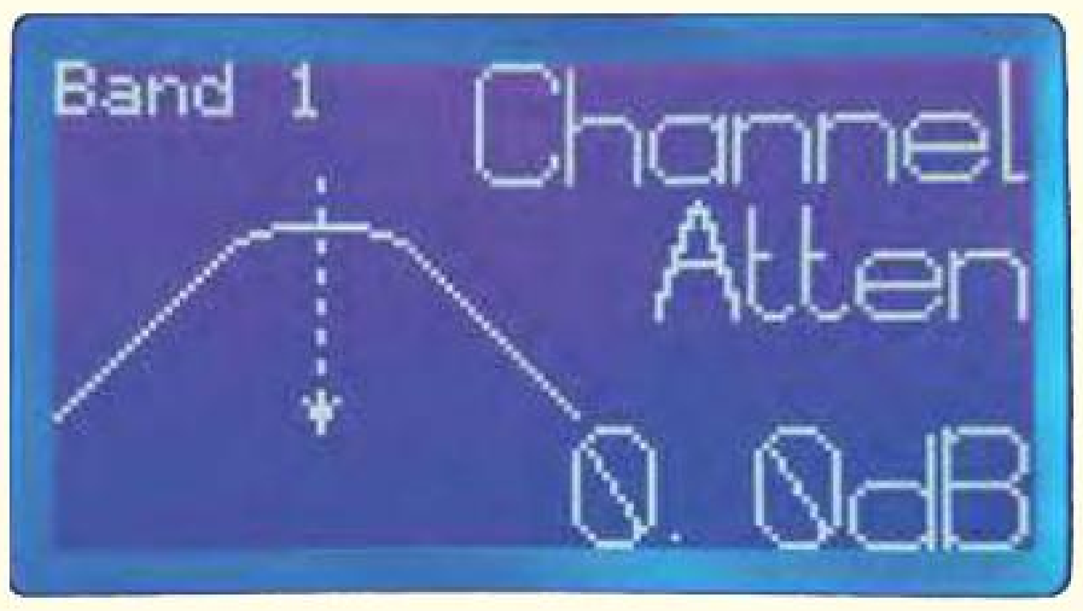
Screen11: the lower slope adjustment. You can select None, 6dB/octave or 12dB/octave Butterworth, or 24dB/octave Linkwitz-Riley filters. Then press S2 to go to…



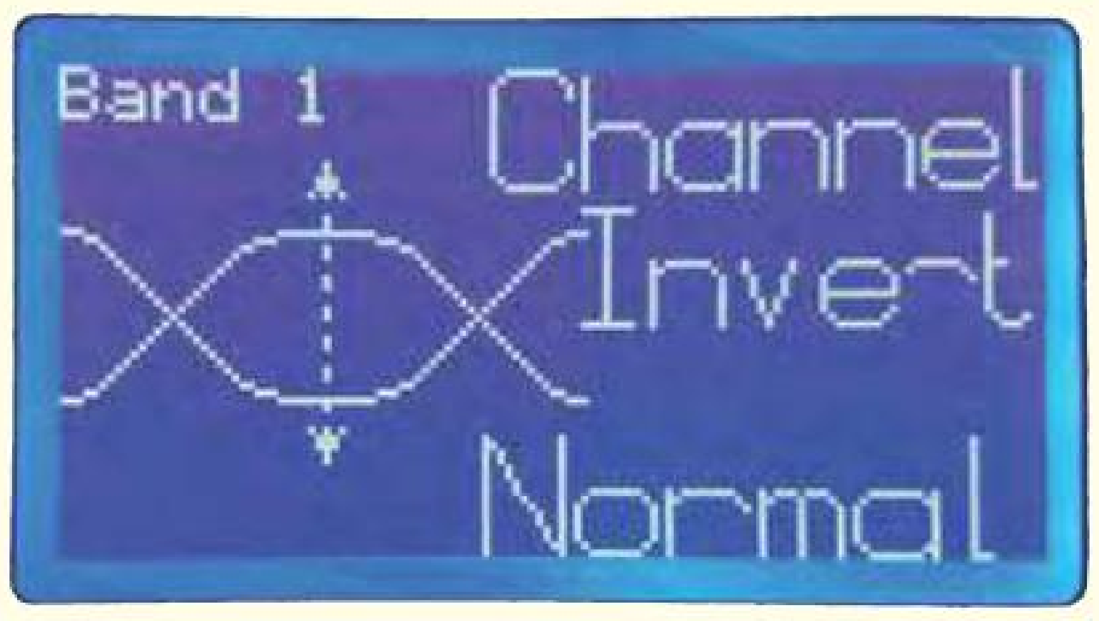
Screen12: the upper slope adjustment, where you have the same options. Press S2 again to go to…



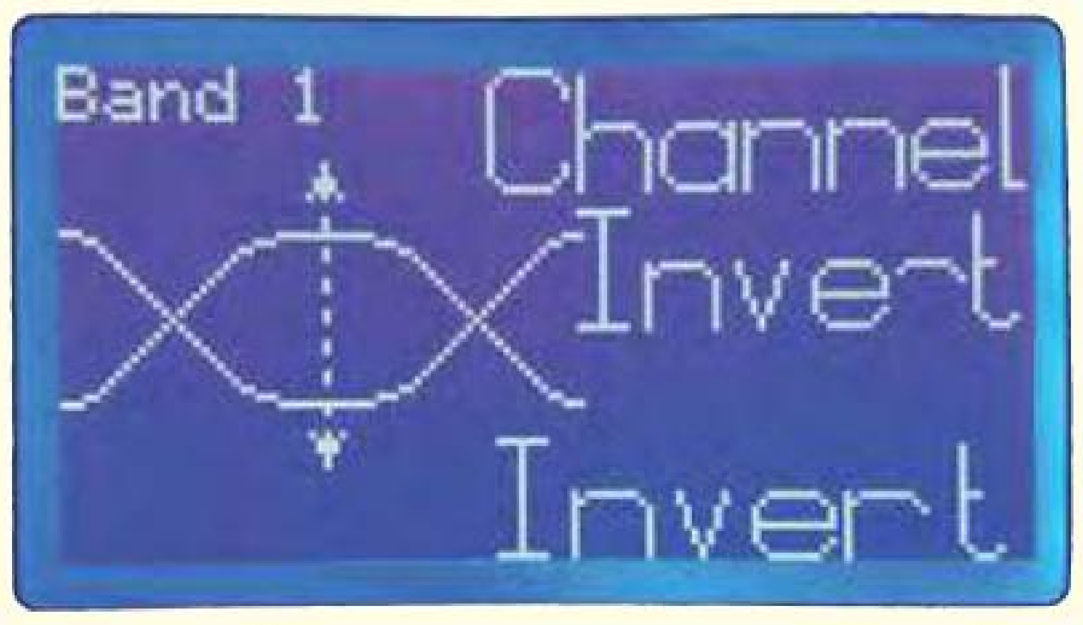
Screen13: the delay adjustment, allowing time-compensation of drivers in a speaker cabinet. The setting (up to 6239mm) is converted to a delay based on the speed of sound. Press S2 again to go to…



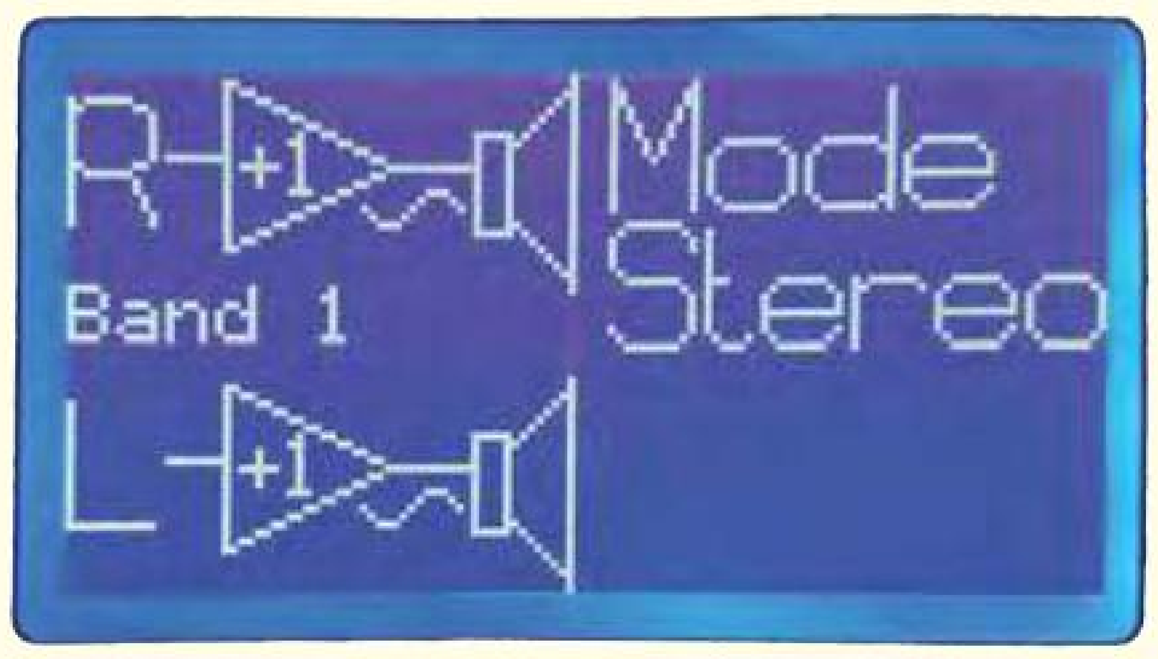
Screen14: the attenuation adjustment, which can be set from 0dB down to −20dB. It can be used to compensate for different driver efficiencies etc. Pressing S2 again takes you to…



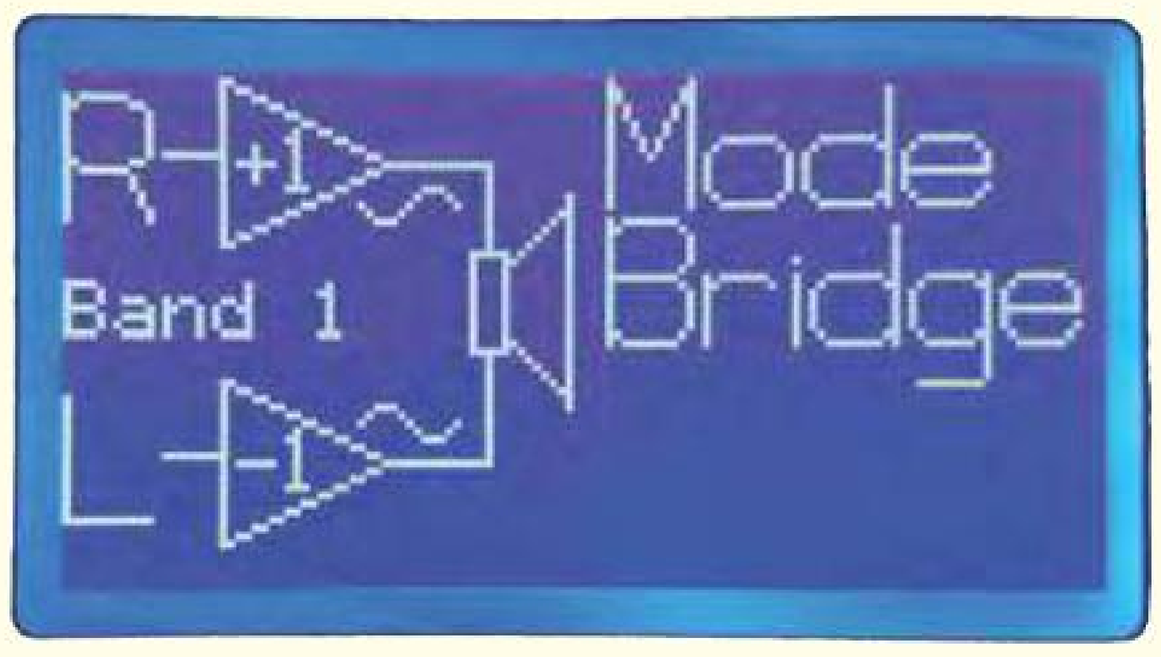
Screen15: the option to invert the signals for this output, which may be useful if you have drivers wired out-of-phase. Rotating the knob…



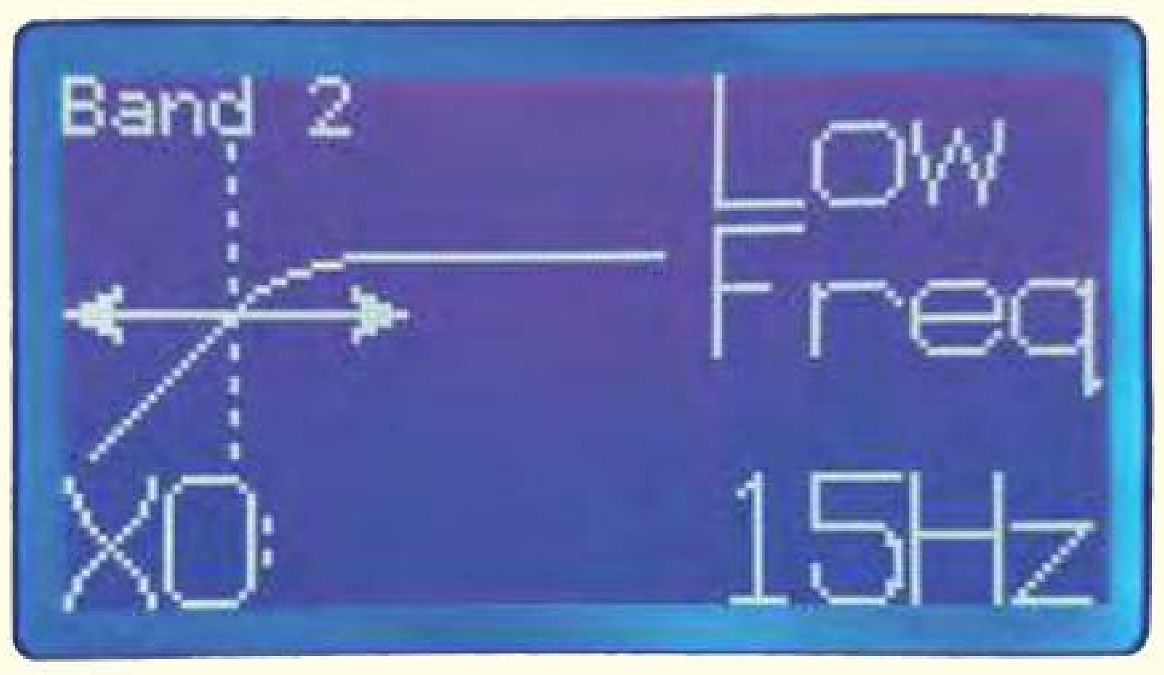
Screen16: selects inverted mode, while rotating it further returns to normal (non-inverted) mode. One more press of S2 takes you to…



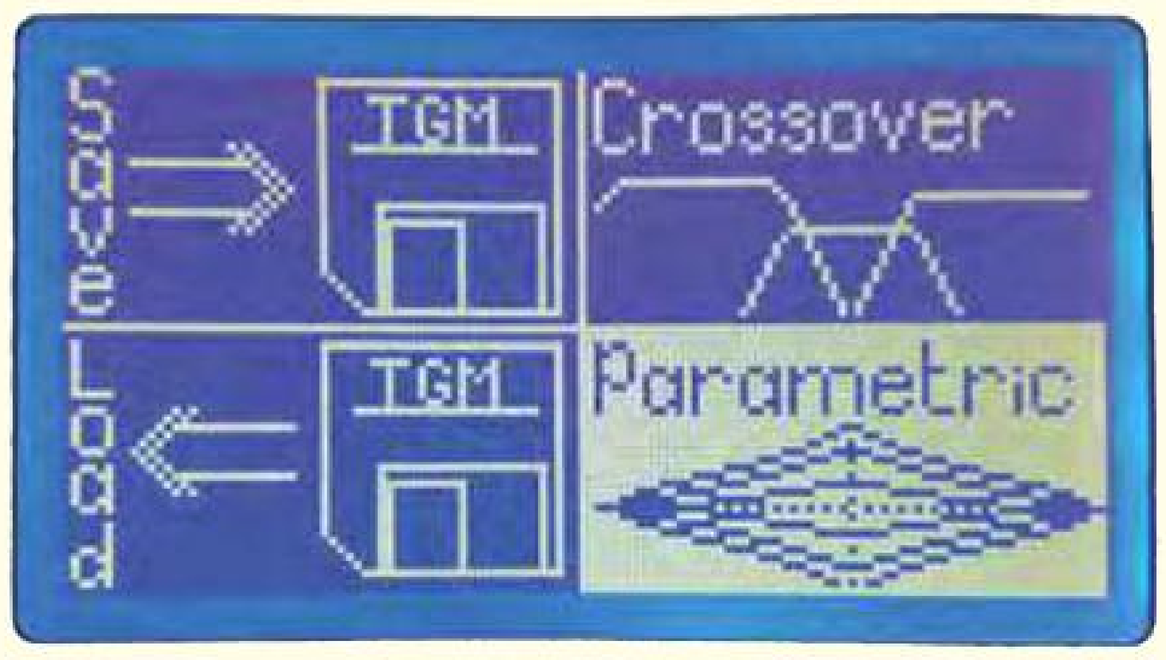
... Screen17: the crossover mode screen. By default, it's 'Stereo', as shown here, but you can change it to…



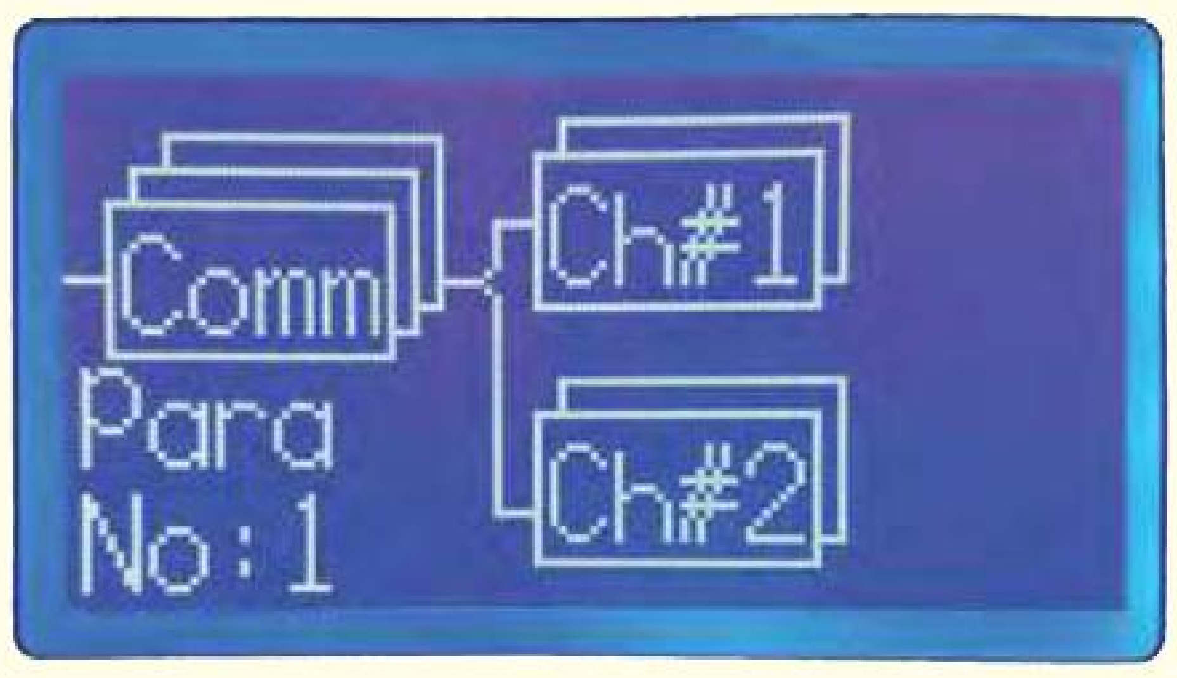
Screen18: Bridge mode, where the second output is an inverted version of the first output, for using two mono amps (or one stereo amp) to drive a speaker in bridge mode. Pressing S2 again…



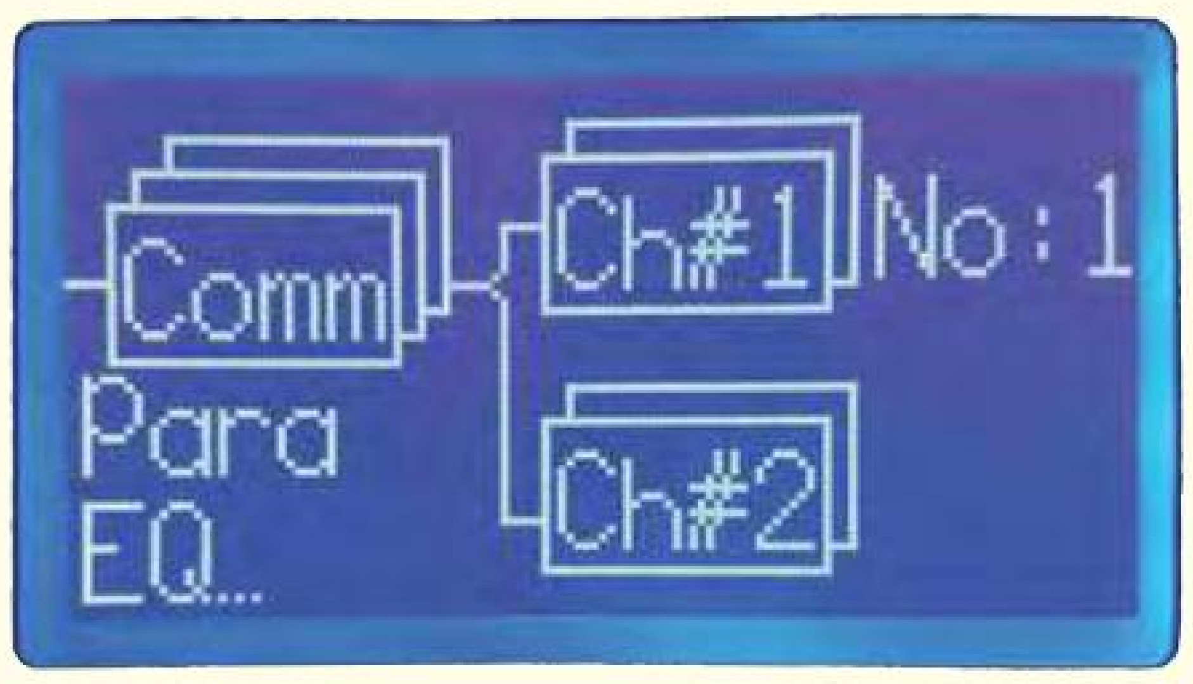
Screen19: cycles through the same set of options for the next band, starting with the lower –3dB point adjustment and then all the different settings and bands until it loops back to Band 1.



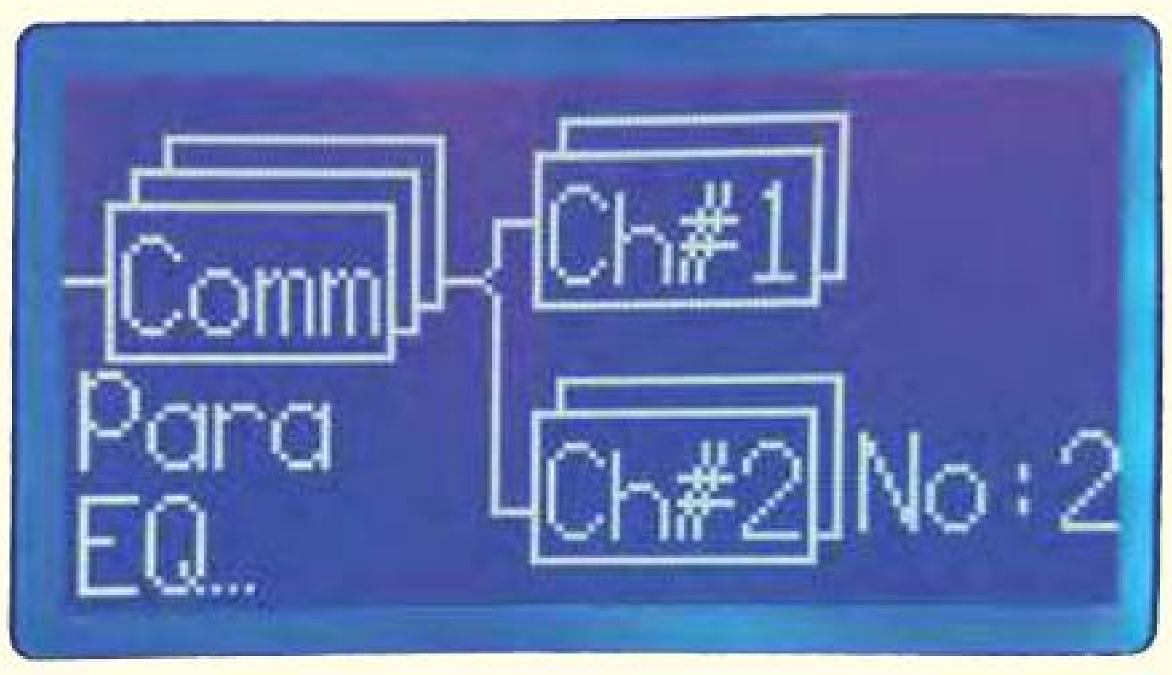
Screen20: here's the main menu again, and this time we have selected the Parametric settings. Pressing S2 takes us to…



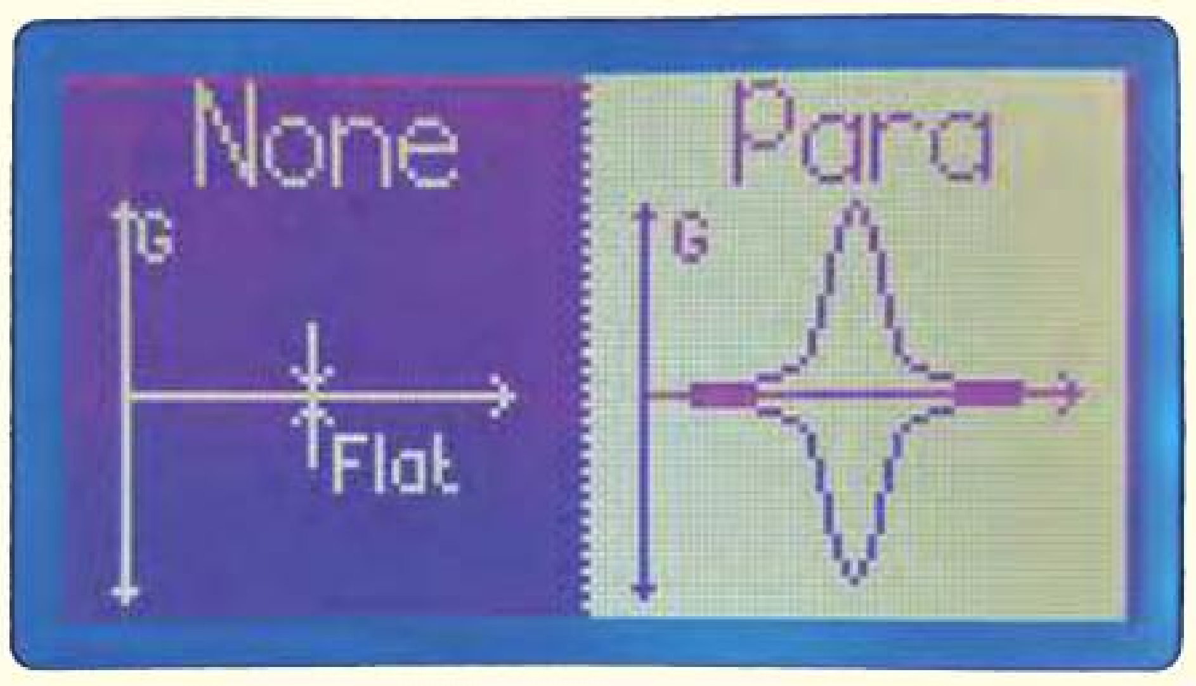
Screen21: this screen lets you choose which parametric equaliser band to adjust. There are four bands which apply to both channels, plus two that only apply to each of the two individual channels….



Screen22: The rotary encoder lets you select any of these eight equaliser bands. Here we have selected the first band, which applies only to Channel 1, and here…



Screen23: we have selected the second band, which applies only to Channel 2. Pressing S2 on any of these options takes you to…



Screen24: this screen, which lets you switch on or off each equaliser band. Pressing S2 again takes you to…

Finally, drill the holes for the rotary encoder, pushbuttons and mounting screws, as shown in Fig.21(a). This can be used as a template, but make sure it's far enough away from the mounting location of the LCD screen that the two boards will not foul each other.

We attached our control board to the rear of the front panel using 9mm M3 tapped nylon spacers, with black machine screws holding it on at the front and nickel-plated machine screws at the rear. Ensure that the holes are large enough to prevent the switches from binding.

You can then attach the rotary encoder knob and connect the LCD panel and control board back to the CPU board and power supply board respectively, as per your earlier tests.

Performance

Fig.20 shows the output of a spectrum analyser connected to one pair of outputs on the *DSP Active Crossover*. A pure 1kHz sinewave is being fed into the inputs. This shows up in the spectral analysis as a large spike just to the left of centre.

The readout below shows that this fundamental signal measures –9.72dBFS for the left channel and 1.62dBFS for the right channel. 'dBFS' stands for 'decibels full scale'. In this case, the full-scale output is around 2.2V RMS, so those signals are at around 0.72V RMS and 1.8V RMS, respectively.

The smaller spikes you can see to the right of the fundamentals, at 2kHz, 3kHz... are the harmonics, ie, the distortion products resulting from the signal passing through the unit. The most significant are at 3kHz and 5kHz, ie, the third and fifth harmonics.

The software measures the relative levels of each harmonic and the fundamental (first harmonic) and feeds them into a formula to calculate the total harmonic distortion (THD) ratio for each channel, which it's showing as 0.0004% for the left channel (remember, that's the one with the reduced signal level!) and 0.0001% for the right channel.

Note that if you incorporated the noise measurement (seen in the wiggly bases of the plots), these figures wouldn't be quite as good, but they're vanishingly low either way, and you certainly won't complain about the sound quality coming out of this device.

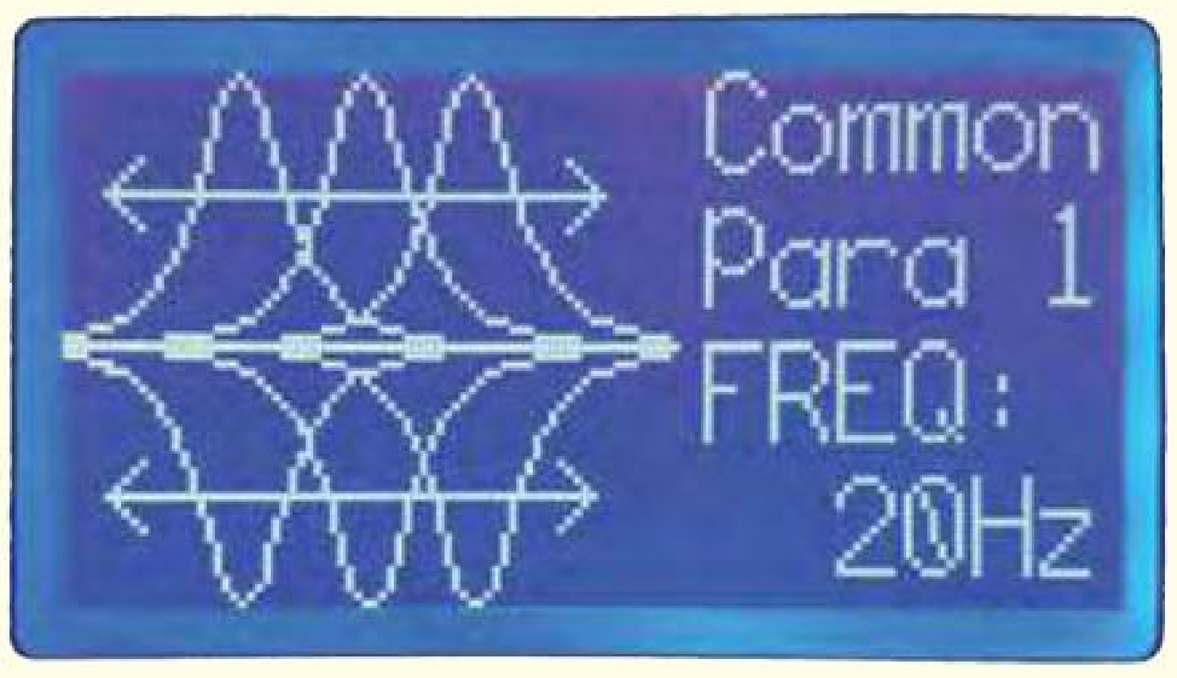
Using it

The *DSP Active Crossover* is set up and controlled using a menu system. Menu entries are shown on the graphical LCD, while the rotary encoder and two pushbuttons are used to scroll through entries, select them and go back to the start. The various menu screens are shown in the panels overleaf and on these pages, along with a description of each one.

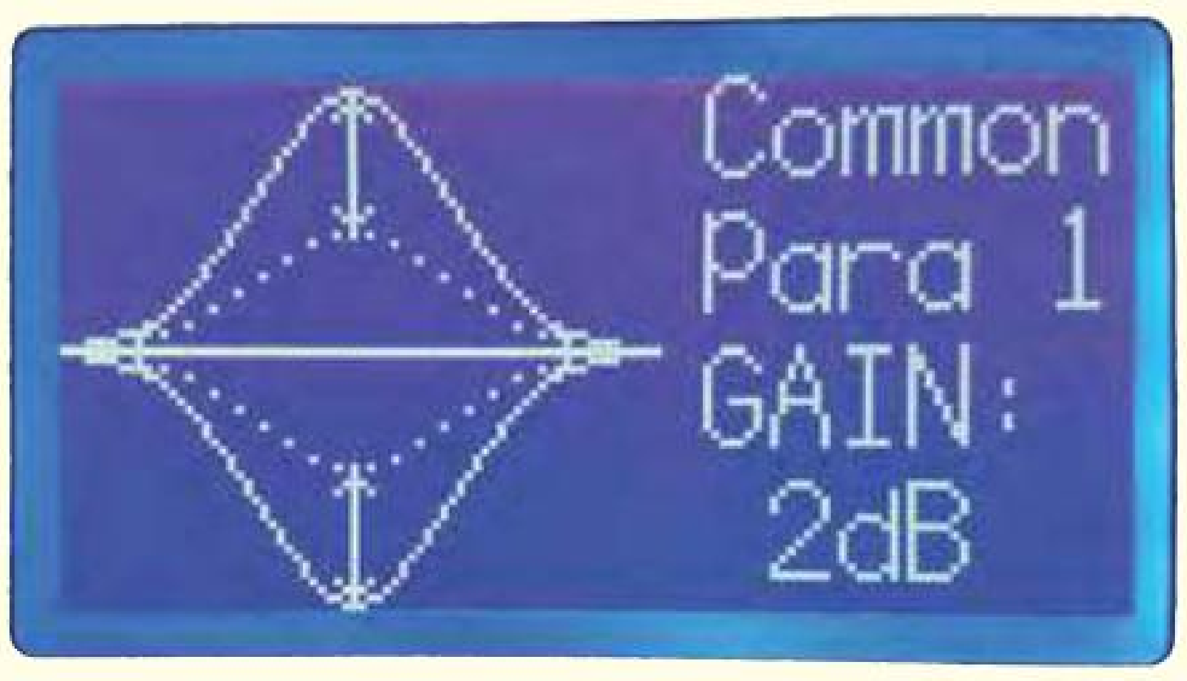
After showing two splash-screens in quick succession, the unit defaults to the volume control screen. This allows you to use it as a preamp, varying the volume with the rotary encoder knob, from –104db up to +12dB (the default is 0dB). Pressing either button (or the knob, if your rotary encoder has an integral button) takes you to the main menu, which has four options.

The rotary encoder selects between those options, while button S2 or the integral rotary encoder pushbutton selects the current option. This button is used as an 'Enter' key while button S1, at right, acts as 'Escape', to go back to the main screen without making any further changes.

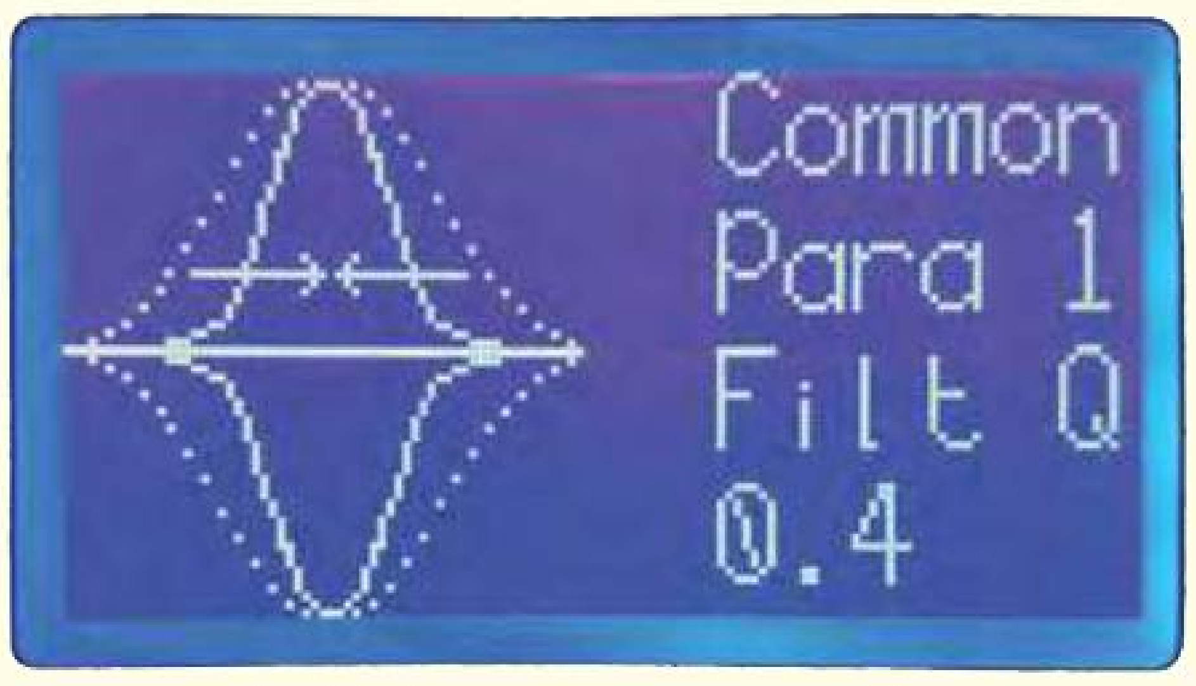
Once you've selected one of the options, you use S2 to cycle through the available sub-options and the rotary encoder to make changes to those options.


Common
Para 1
FREQ:
20Hz

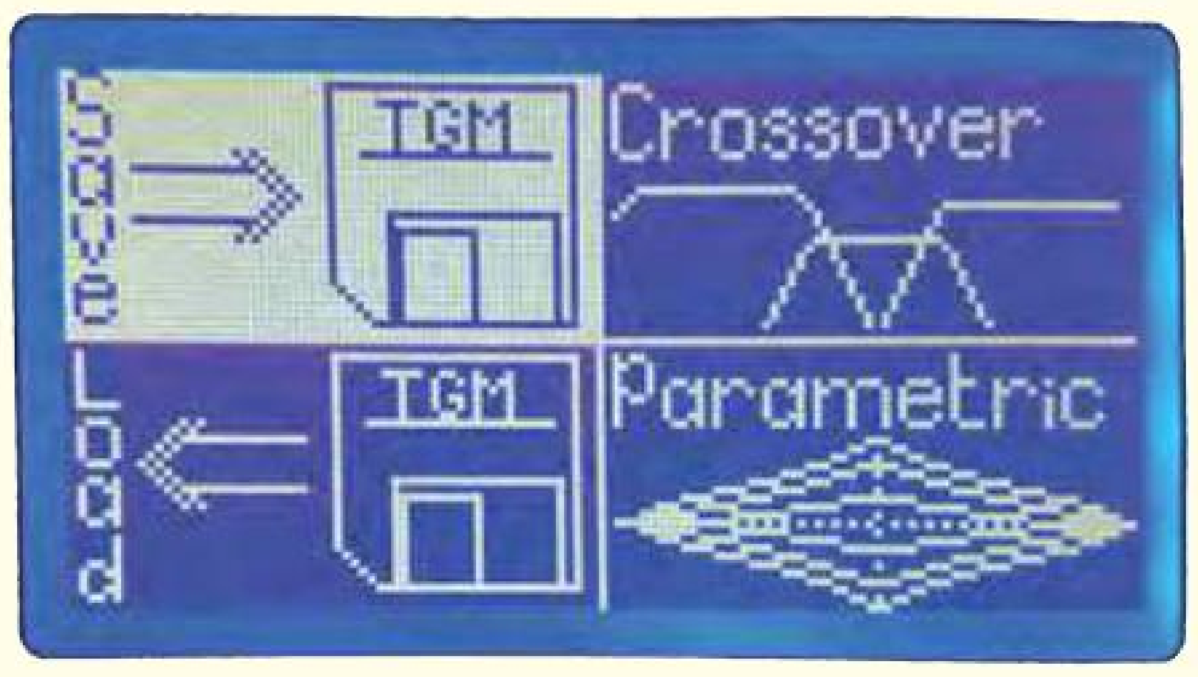
... Screen25: the centre frequency adjustment screen. Select a frequency from 15Hz to 15kHz using the rotary encoder, then press S2 to…


Common
Para 1
GAIN:
2dB

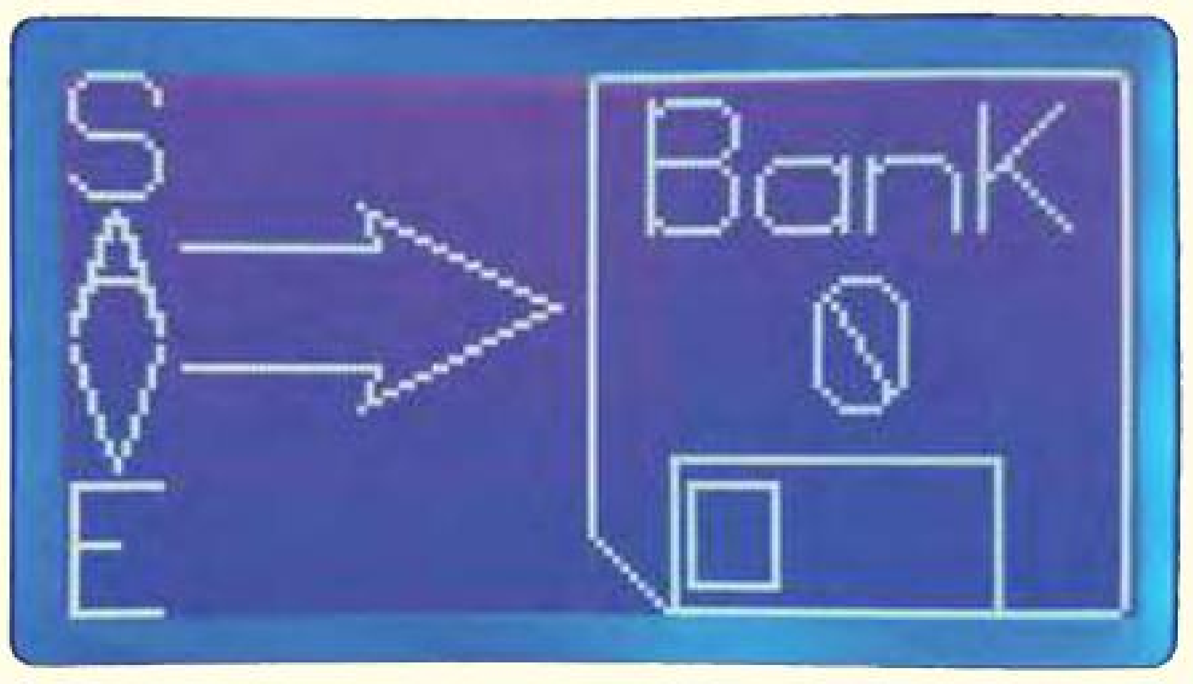
Screen26: adjust the gain or cut for this equaliser band, from –10dB to +10dB. Pressing S2 again…


Common
Para 1
Filt Q
0.4

Screen27: lets you set the Q of the filter, to a value between 0.1 and 10, which affects how wide a range of frequencies it affects.

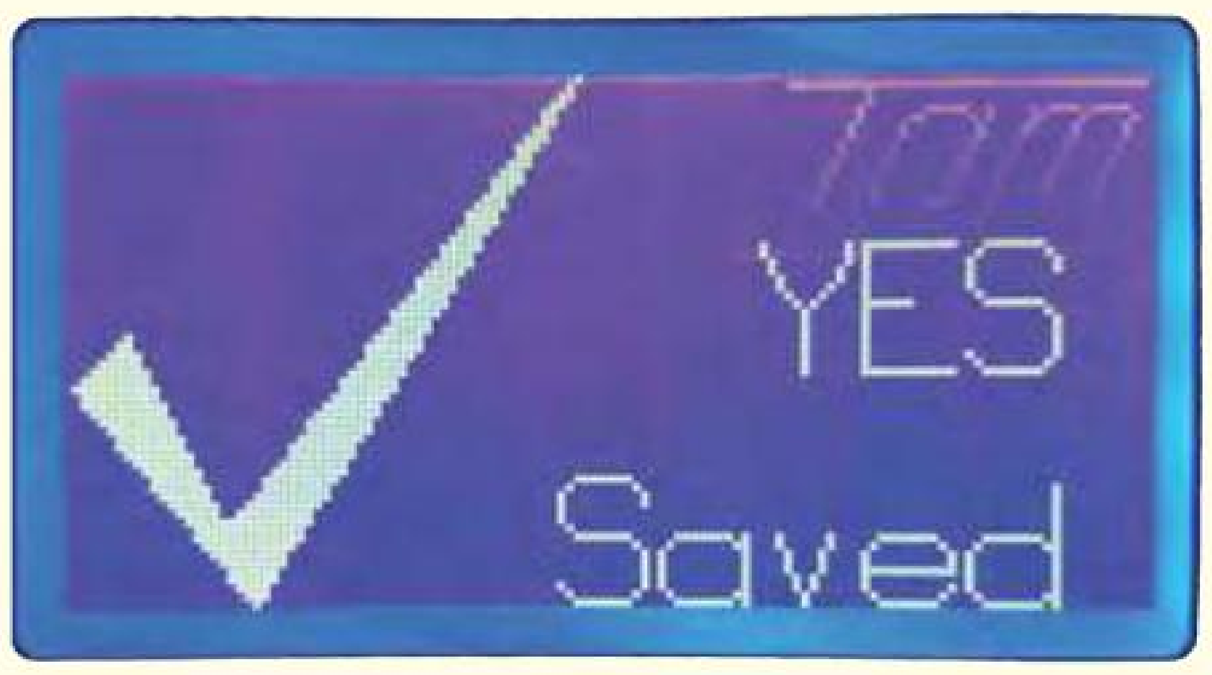


Screen28: back at the main menu, this time we've selected the Save option. Pressing S2 brings us to…



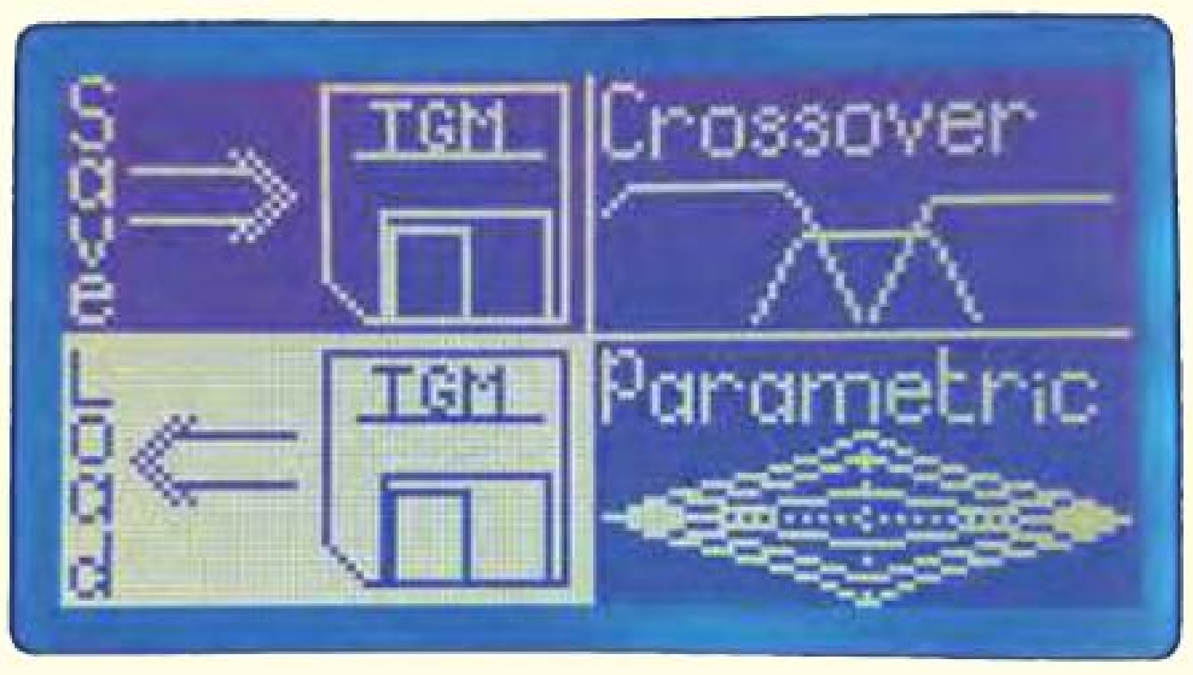
Bank
0

Screen29: a screen where you can choose one of three settings banks to save to. Use the rotary encoder to select one, or press S1 to abort. Press S2 or the knob…

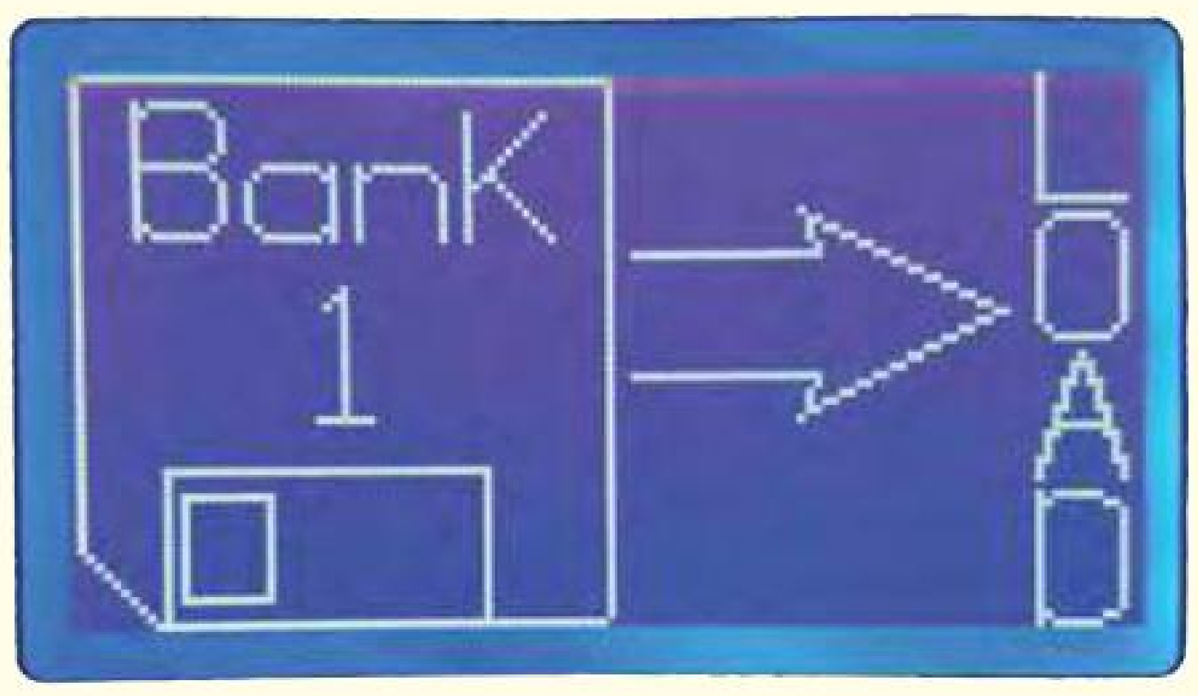


YES
Saved

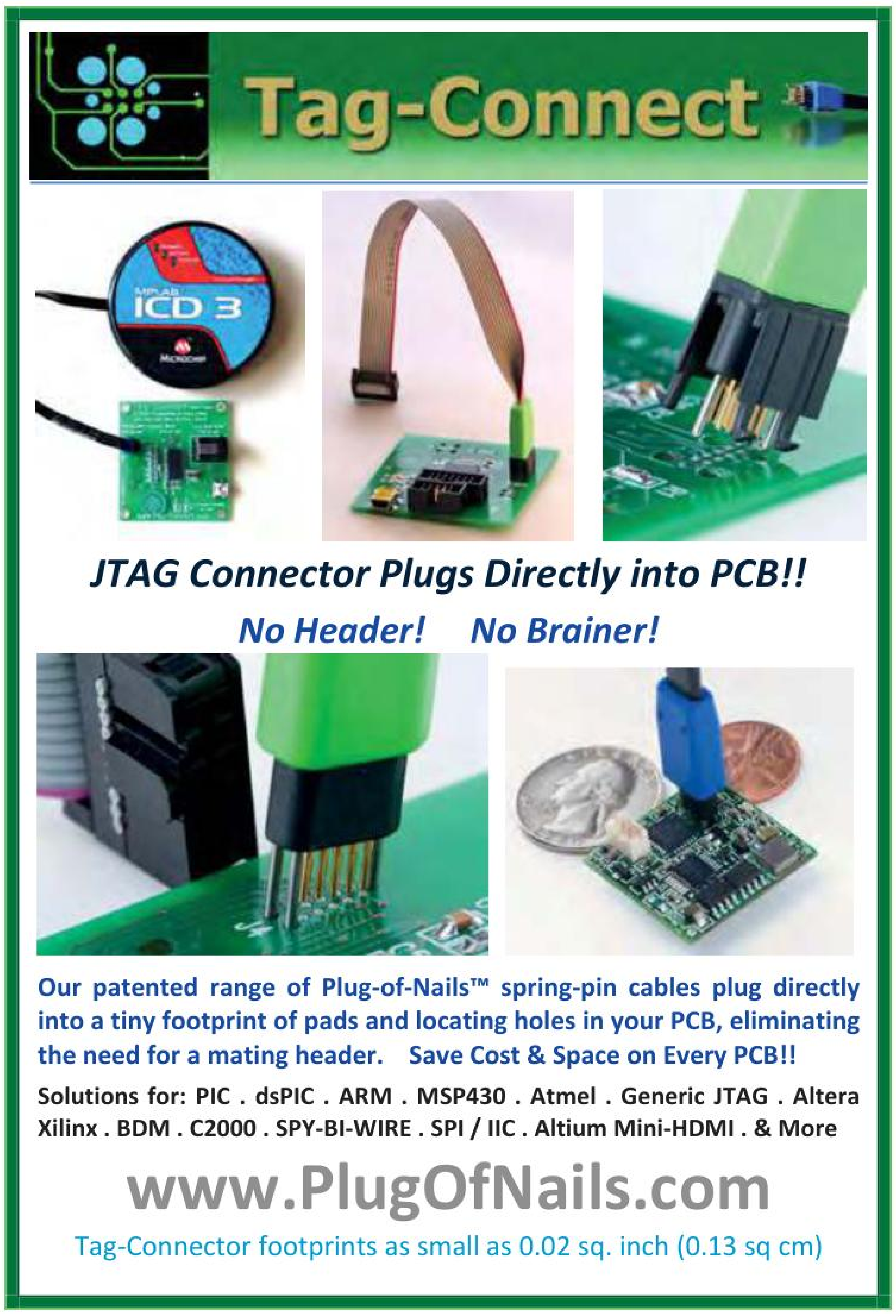
Screen30: to save the settings to EEPROM. This screen is displayed for a short time, then the display returns to the default screen, ie, volume control mode (Screen03).



Screen31: the final main menu option loads the settings you've saved. Bank 0 loads by default at start-up. To load a different configuration, select this option and press S2…


Bank
1

Screen32: then select a bank to load using the rotary encoder, and either press S2 to load it, or S1 to abort and go back to the volume control screen.


Tag-Connect

JTAG Connector Plugs Directly into PCB!!

No Header! No Brainer!

Our patented range of Plug-of-Nails™ spring-pin cables plug directly into a tiny footprint of pads and locating holes in your PCB, eliminating the need for a mating header. Save Cost & Space on Every PCB!!

Solutions for: PIC . dsPIC . ARM . MSP430 . Atmel . Generic JTAG . Altera Xilinx . BDM . C2000 . SPY-BI-WIRE . SPI / IIC . Altium Mini-HDMI . & More

www.PlugOfNails.com

Tag-Connector footprints as small as 0.02 sq. inch (0.13 sq cm)

ESR Electronic Components Ltd

All of our stock is RoHS compliant and CE approved. Visit our well stocked shop for all of your requirements or order on-line.

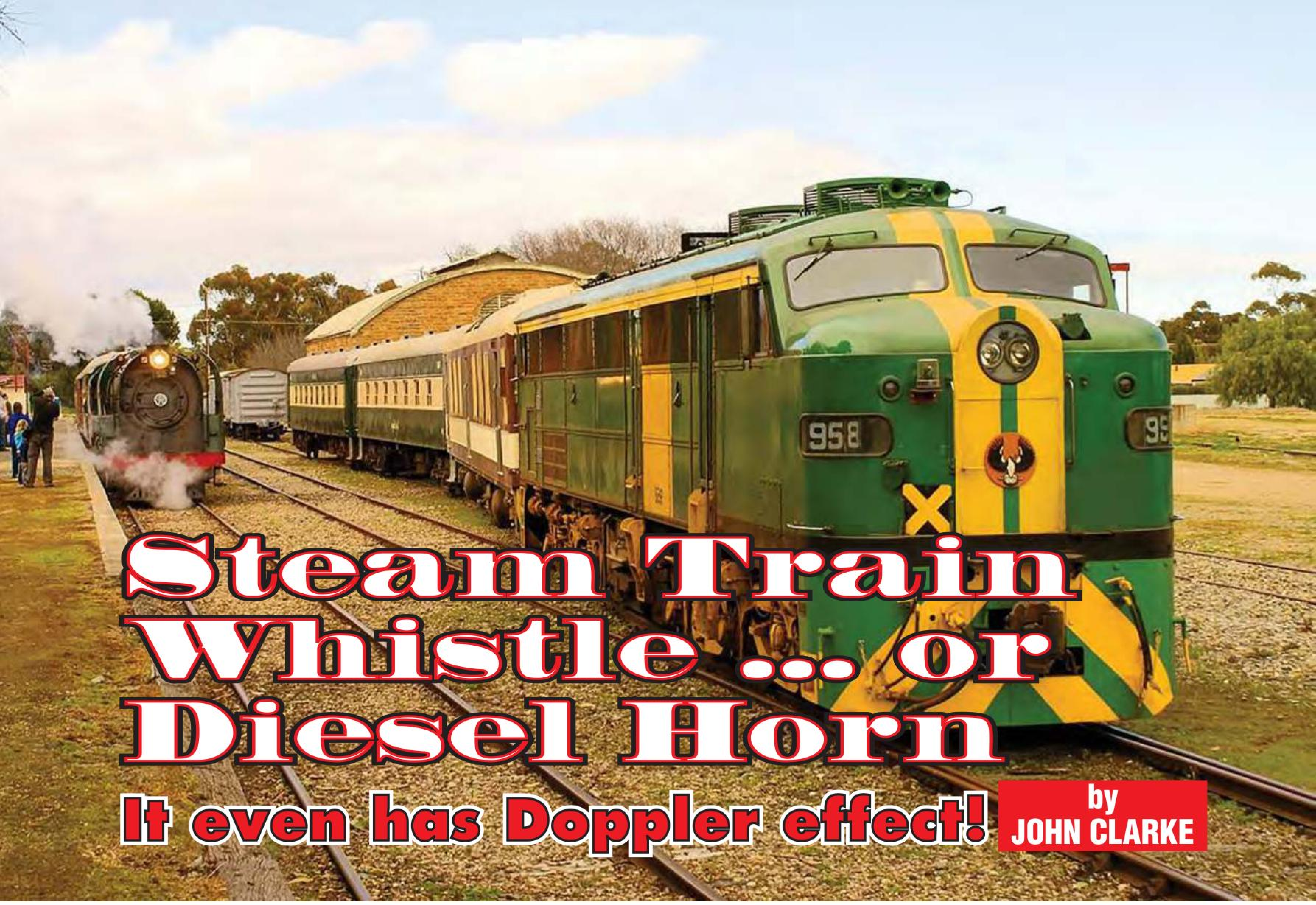
We can help and advise with your enquiry, from design to construction.

3D Printing • Cable • CCTV • Connectors • Components • Enclosures • Fans • Fuses • Hardware • Lamps • LED's • Leads • Loudspeakers • Panel Meters • PCB Production • Power Supplies • Relays • Resistors • Semiconductors • Soldering Irons • Switches • Test Equipment • Transformers and so much more…

Monday to Friday 08:30 - 17.00, Saturday 08:30 - 15:30

Station Road
Cullercoats
North Shields
Tyne & Wear
NE30 4PQ

Tel: 0191 2514363 sales@esr.co.uk www.esr.co.uk

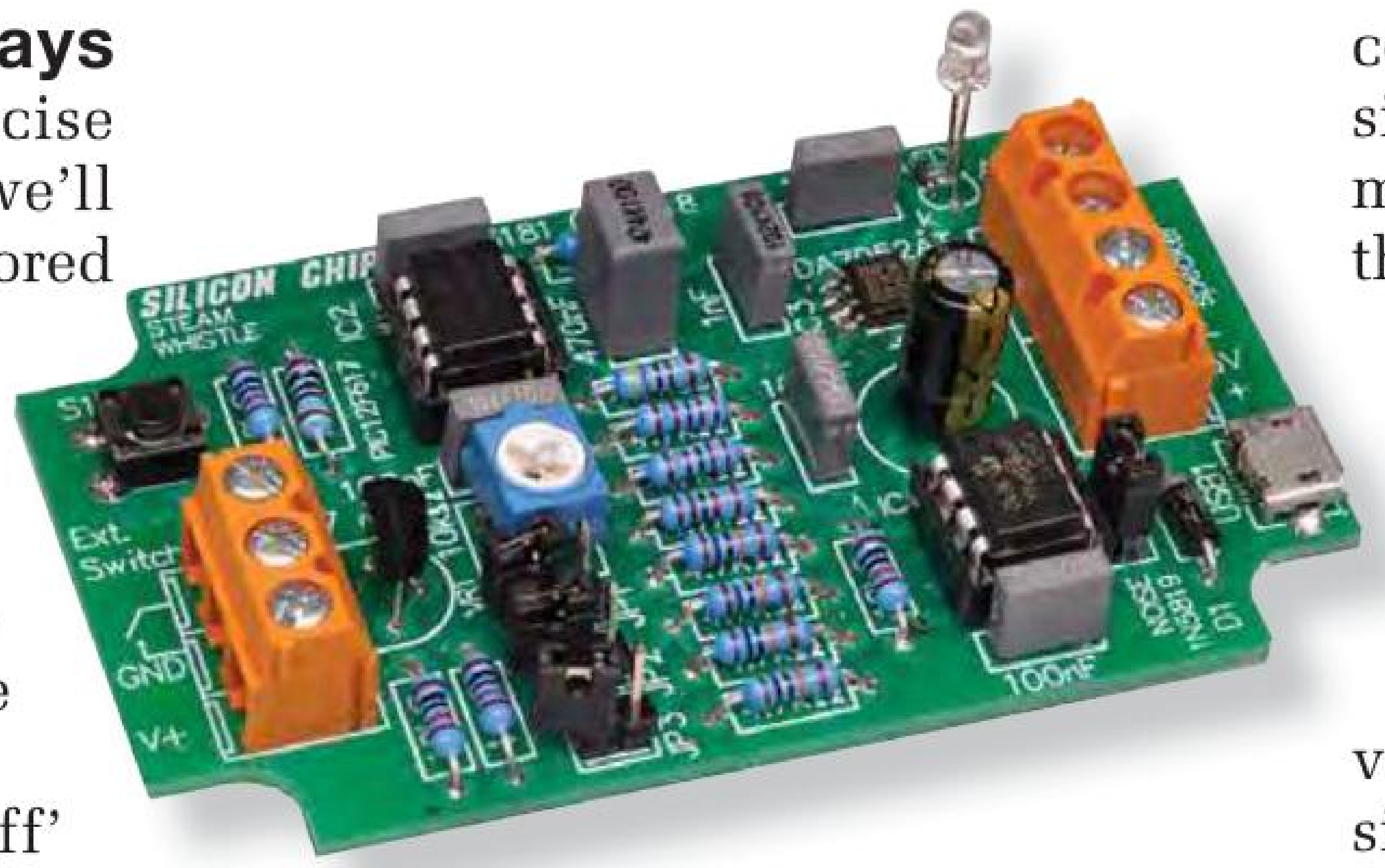


Steam Train Whistle ... or Diesel Horn

It even has Doppler effect!

by
JOHN CLARKE

Relive the exciting days of steam-train travel with this *Steam Train Whistle / Diesel Horn Sound Generator*. Use it in your model railway layout, as a doorbell or just as a standalone sound effect. It can even simulate the Doppler effect, providing a change in pitch as if the train is passing by. Plus, the Whistle/Horn sound can be customised in a number of ways, to suit your project's preferences.



Steam trains are always popular (well, to be precise steam *locomotives*, but we'll stick to 'trains' here!) Many restored trains can be seen travelling the countryside at heritage rail lines. Their popularity is proven by the crowds of people gathering to watch alongside the track and the number of people enjoying the ride.

Along with the 'chuff-chuff' sounds of the steam engine, it is the whistle that gives the most excitement and nostalgia.

It is the toot of the whistle as the train departs; it is the sound of the whistle as it passes you by and the blast of the whistle as the train enters a tunnel or approaches a level crossing.

The great thing about this circuit is that you can get that steam train sound without the corresponding sooty face! Our device simulates a steam-powered whistle by mixing three separate oscillators with the output of a white noise generator.

These oscillators generate the whistle chimes, with plenty of harmonics to ensure they have a rich sound, while the noise generator provides the sound of the steam rushing out.

You can adjust the rise time of the volume at the start of the whistle. That simulates the rate at which the cord is pulled to open the steam valve to the whistle.

At the end of the whistle period, when the cord is released, the valve quickly shuts itself off due to steam pressure (in our case, in 100ms).

If you elect to use Doppler effect simulation, the situation is somewhat different. In this mode, the rise in volume simulates the train whistle starting from a

Features

- Creates steam whistle or diesel horn sound effects
- Steam simulation using white noise
- Adjustable volume rate rise for the steam whistle
- Multiple trigger options
- Adjustable whistle time
- Optional Doppler effect
- Adjustable whistle frequencies

distance away and then increasing in volume as it approaches nearer.

The decay in volume after the train passes (and after the frequency shifts) simulates the decrease in volume as the train moves away.

The whistle/horn sound can be initiated with a manual pushbutton, a microswitch, reed switch, relay or by a signal from an external microcontroller.

Design

There have been many electronic steam engine effect designs; this new one boasts many novel features, including Doppler effect, despite using a simpler circuit. That's because it is instead based around an inexpensive PIC microcontroller.

The basic concept is as follows. The micro generates three different pulse trains, each with a fixed duty cycle and adjustable frequency.

These pulse trains contain various frequencies including the fundamental and higher harmonics, which when mixed together, make a whistle or horn sound. When simulating a steam whistle, they are also mixed with white noise, as explained above.

The mixed sound is then fed to an audio amplifier with a built-in volume control circuit. The volume control signal comes from the same microcontroller which is producing the whistle/horn sounds. The amplifier then drives a speaker to produce the final sound.

The length of the whistle/horn sound can be adjusted. If set to minimum, the whistle sound will only be for as long as the trigger switch is closed. Or it can be set to a longer time so that a brief press of the switch will initiate the sound for a fixed period. Depending on the mode, holding the switch down may or may not extend the sound beyond this set period.

Circuit description

The circuit is shown in Fig.2 and comprises three ICs: two PIC12F617 microcontrollers (IC1 and IC2) and a TDA7052A 1W Bridge-Tied-Load (BTL) mono audio amplifier with DC volume control (IC3).

IC1 is the white noise source and runs an internal program that generates noise over the full audio spectrum. You'll find a full description of its

5-chime and 3-chime steam whistles and diesel horns

The steam whistle featured in this article has three oscillators controlled by the microcontroller. This is ideal for simulating the sound of 3-chime steam whistles but many other steam locomotives had 5-chime whistles.

Clearly, we could have designed the circuit with five oscillators but that would have a required a microcontroller with more pins and more passive components. However, since the simulated sound of this circuit is really quite convincing, we think this is a reasonable compromise.

If you want the exact whistle for a particular locomotive, it would better to use our Super Digital Sound Effects Module (see August and September 2019 issues of PE) together with the correct WAV stored on its microSD card.

These comments also apply to diesel locomotive horns. Some large diesel locomotives have 5-chime horns but many did have 3-chime or even 2-chime units.

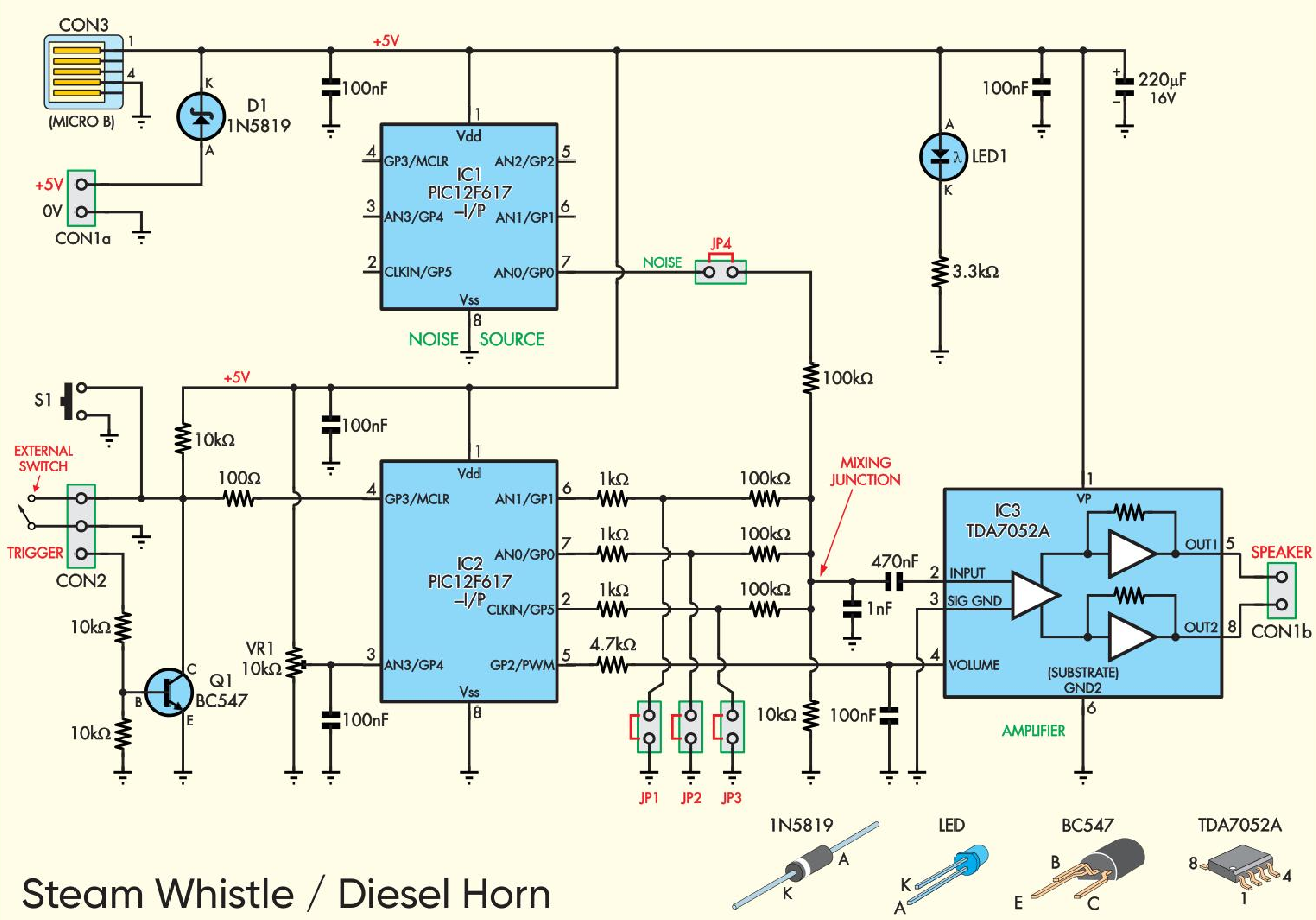

CON3
(MICRO B)
+5V
D1
1N5819
K
A
+5V
0V
CON1a
100nF
IC1
PIC12F617
–I/P
Vdd
GP3/MCLR
AN3/GP4
CLKIN/GP5
AN2/GP2
AN1/GP1
AN0/GP0
Vss
NOISE SOURCE
NOISE
JP4
100nF
220µF
16V
LED1
3.3kΩ
100kΩ
S1
EXTERNAL SWITCH
TRIGGER
CON2
10kΩ
100Ω
Q1
BC547
B
C
E
VR1
10kΩ
IC2
PIC12F617
–I/P
CLKIN/GP5
GP2/PWM
1kΩ
4.7kΩ
100kΩ
MIXING JUNCTION
470nF
1nF
JP1
JP2
JP3
IC3
TDA7052A
VP
INPUT
SIG GND
VOLUME
OUT1
OUT2
(SUBSTRATE) GND2
AMPLIFIER
SPEAKER
CON1b
1N5819
LED
BC547
TDA7052A
Steam Whistle / Diesel Horn

Fig.1: IC1 produces white noise which is used to emulate steam sounds, while IC2 produces three pulse trains which are mixed together to give a whistle or horn sound. This signal is then fed to amplifier IC3 which drives the speaker. It is powered from a 5V DC or USB supply, and the sound is triggered by onboard switch S1 or an external switch or signal via CON2.

Specifications

• Power supply	5V at 300mA from an external supply via screw terminals or USB cable
• Output power	about 1W into 8Ω
• Regular whistle period	100ms to 22.5s, extended if S1 held down (no Doppler effect) 2.5s to 22.5s, no extension (with Doppler effect)
• Whistle volume rise time	5ms to 8s
• Whistle volume fall time	100ms fixed (no Doppler effect) 5ms to 8s (with Doppler effect)
• Whistle frequency range:	244Hz to 1.053kHz
• Frequency adjustment steps	3Hz at 250Hz, 6Hz at 333Hz, 12Hz at 500Hz, 24Hz at 666Hz, 48Hz at 1kHz
• Simulated Doppler speed	80km/h

operation in the *White Noise Generator* project published in the September 2019 issue of *PE*. It was designed so that it can be re-used in other circuits where a white noise source may be required. For now, all you need to know is that a white noise 'hissing' audio signal is produced at its pin 7 output.

This noise signal is fed to the audio mixing point, at the junction of the 10kΩ resistor and 100kΩ resistor, via JP4, which can be used to disconnect the white noise source when simulating a diesel horn or while adjusting the oscillator frequencies.

IC2 produces the whistle and horn sounds. It does this using three pulse trains from its GP1, GP0 and GP5 digital outputs (pins 6, 7 and 2).

It also produces a DC control signal by filtering a PWM signal which is produced at pin 5. This is fed to the volume control input on amplifier IC3 and this is used as an 'envelope' for the whistle and horn sounds.

IC2 also monitors an external switch via its GP3 input (pin 4, which is used to trigger the sound effects. And it reads the wiper position of trimpot VR1 using its AN3 analogue input (pin 3), which controls various options such as the component frequencies of the whistle/horn sound, the whistle/horn time and the volume rise rate for the steam whistle.

The pulse trains from the GP1, GP0 and GP5 outputs are fed to an audio mixing point, at the junction of the 100kΩ and 10kΩ resistors, via a 1kΩ series resistor for each output.

The pulse trains from those three outputs are not quite square waves; a square wave has a duty cycle of 50% while the pulses from these outputs have a duty cycle of 43.75%.

This provides a richer set of harmonics than a square wave. A square wave has only odd harmonics whereas this series of slightly shorter pulses also has even harmonics, which produces a more realistic sound.

The supersonic harmonics of the mixed signal are filtered out by a 1nF capacitor to ground and the audio signal is then AC-coupled to input pin 2 of audio amplifier IC3 via a 470nF capacitor. This drives the speaker connected to CON1 in bridge mode.

This chip has a volume control input on pin 4. A DC voltage is applied to this pin and the higher the voltage, the louder the output volume.

This volume control signal is produced by microcontroller IC2 using filtered PWM, from its pin 5 PWM output. The 4.7kΩ resistor and 100nF capacitor form an RC low-pass filter with a –3dB point of 339Hz.

The actual PWM frequency is 19.61kHz, which is so much higher than the filter corner frequency that the output of the filter is effectively a DC voltage, proportional to the PWM duty cycle.

Initially, the pin 5 output is low, setting the attenuation in IC3 to its maximum value of more than –70dB, which mutes the audio output. The PWM pulse width is gradually increased to bring the average voltage at pin 4 up to 1V, resulting in an eventual amplifier gain of +20dB.

The steam whistle/horn sound effect is triggered by pressing pushbutton S1 to pull the GP3 input (pin 4 of IC1) low. It is normally held high by a 10kΩ resistor to the 5V rail.

You can also use an external switch between pins 1 and 2 of CON2 (which could be a set of relay contacts) or by applying at least 1.5V between pins 2 and 3 of CON2, which switches on NPN transistor Q1, pulling the GP3 input low.

The circuit is powered from 5V, either applied to part of terminal block CON1 or via a micro type-B USB connector, CON3.

When using CON1, diode D1 provides reverse-polarity protection. The supply is bypassed using a 220µF capacitor and several 100nF capacitors, one for each IC.

Changing the sound

Jumper shunts JP1-JP3 allow the frequencies of these pulse trains to be adjusted. Two of these jumpers are inserted at a time, shorting two of the pulse trains to ground and thus disabling them. This allows you to measure the frequency of the third pulse train and make adjustments using VR1. Pressing pushbutton S1 then saves the new frequency setting to EEPROM.

Microcontroller IC2 detects whether any of the jumpers are inserted at start-up by enabling internal pull-up currents for the three output pins and then sensing whether any of them are held at ground potential.

If so, it goes into adjustment mode. If none of the jumpers are present, all three outputs will be high and the software goes into the normal sound -effects generation mode.

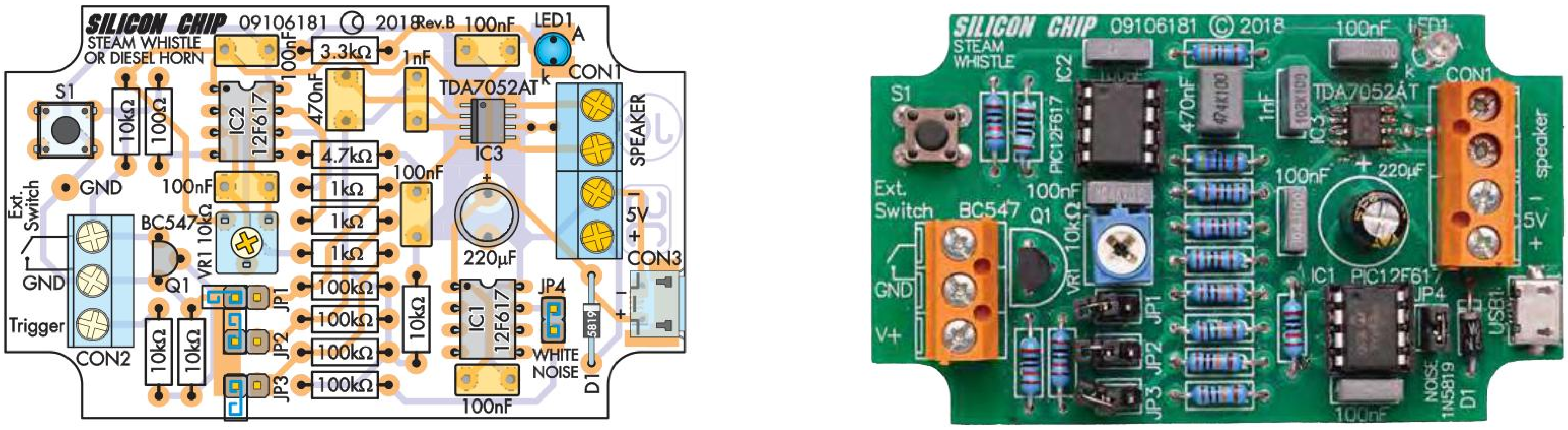


Fig.2: use this PCB overlay diagram and matching photo as a guide for building the *Steam Whistle / Diesel Horn* board. IC3 and CON3 are the only SMD components. Take care when soldering CON3 since the pins are small and close together, and easy to bridge. Don't get IC1 and IC2 mixed up; while they are the same type of chip, they are programmed with different firmware.

Parts list – Steam Whistle/Diesel Horn

1 double-sided PCB, coded 09106181, 79 x 48mm, available from the PE PCB Service
2 2-way screw terminals with 5.08mm pin spacing (CON1)
1 3-way screw terminal with 5.08mm pin spacing (CON2)
1 8Ω 1W loudspeaker [eg, Jaycar AS3030 or AS3004, Altronics C0603C]
1 PCB-mounting micro type-B USB socket (CON3) [Altronics P1309, Jaycar PS0922]
2 8-pin DIL IC sockets (for IC1,IC2)
1 momentary pushbutton switch (S1) [Altronics S1120, Jaycar SP0600]
4 2-way headers with 2.54mm spacings with shorting blocks (JP1-JP4)
1 PC stake

Semiconductors
1 PIC12F617-I/P programmed with 0910618A.HEX (IC1)
1 PIC12F617-I/P programmed with 0910618M.HEX (IC2)
1 TDA7052AT/N2 1W BTL DC volume control amplifier (IC3) [Cat SC3551]
1 BC547 NPN transistor (Q1)
1 1N5819 1A schottky diode (D1)
1 3mm LED (LED1)

Capacitors
1 220µF 16V PC electrolytic
1 470nF 63V or 100V MKT polyester
5 100nF 63V or 100V MKT polyester
1 1nF 63V or 100V MKT polyester

Resistors (all 1%, 0.25W)
4 100kΩ 4 10kΩ 1 4.7kΩ 1 3.3kΩ 3 1kΩ 1 100Ω
1 10kΩ mini horizontal trimpot, code 103 (3386F style) (VR1)

A similar method is used to adjust the initial volume ramp rate for the horn/whistle. This is done by inserting all three shunts, rotating VR1 and then pressing S1 to store the new ramp time.

Trimpot VR1 connects across the 5V supply and so its wiper voltage sweeps from 0V to 5V as its screw is rotated clockwise.

This voltage is applied to the AN3 analogue input of IC2 and converted to a number using its internal analogue-to-digital convert (ADC). It is used to set the pulse train frequencies and initial volume ramp rate as described above

Construction

The *Steam Train Whistle* is built on a double-sided PCB coded 09106181, measuring 79 x 48mm, available from the *PE PCB Service*. It can be housed in a UB3 plastic utility box if desired.

The overlay diagram, Fig.2, shows where the parts are fitted. Install SMD IC3 first as it is easier to solder the pins when there are no components surrounding it.

You will need a fine-tipped soldering iron and 0.7mm diameter solder.

First, align the IC pins onto the pads making sure the that chamfered side of the chip is positioned towards the pin 1 indicator on the board, as shown in Fig.2. Tack solder one of the outside pins to its pad and check that the IC alignment is correct. Re-melt the solder and realign if necessary. Then solder the remaining pins. Make sure you refresh the solder on the first pin at the end. If you accidentally bridge two pins, the excess solder can be removed with a dab of flux paste and some solder wick.

The micro type-B USB connector is soldered in a similar manner to IC3. Align the leads to the pads and solder the two outer flanges on the sides of the USB housing first, followed by the five pins. Clear any pins that are shorted with solder wick.

The resistors – do use a digital multimeter to check the values of each resistor before soldering it as the colour codes can be mistaken.

Mount diode D1 next, with the striped end (cathode) oriented as shown in the overlay diagram.

We recommend using an IC socket for IC1 and IC2. Take care with orientation when installing the sockets – use Fig.2 as a guide.

Now fit headers for jumpers JP1-JP4 and the PC stake at the GND position. Follow with the capacitors, starting with the smaller MKTs and then the 220µF electrolytic capacitor. This capacitor is polarised and must be installed with the polarity shown, with the longer lead through the hole marked '+' (the striped side is negative).

Install transistor Q1, switch S1 and trimpot VR1. Then you can mount terminal blocks CON1 and CON2. CON1 comprises two dovetailed 2-way screw connectors and CON2 is a 3-way screw connector.

Ensure that all the terminal blocks are fitted with the wire entry holes to the outside edge of the PCB.

Finally, LED1 can be soldered in place. We mounted it with the plastic lens 15mm above the PCB so that it would protrude through the lid of the UB3 Jiffy box, but you could mount it at a different height if necessary. Fit it with the longer anode lead soldered to the pad marked 'A' on the PCB.

Although shown as a bare board, the *Steam Train Whistle* can be installed in a UB3 box. The PCB clips it into the moulded side rails on the inside of the box. Cutouts can then be made in the ends of the box for the USB connector and the wires going to screw terminals CON1 and CON2. The loudspeaker is ideally mounted in a small box so that it has a good bass response.

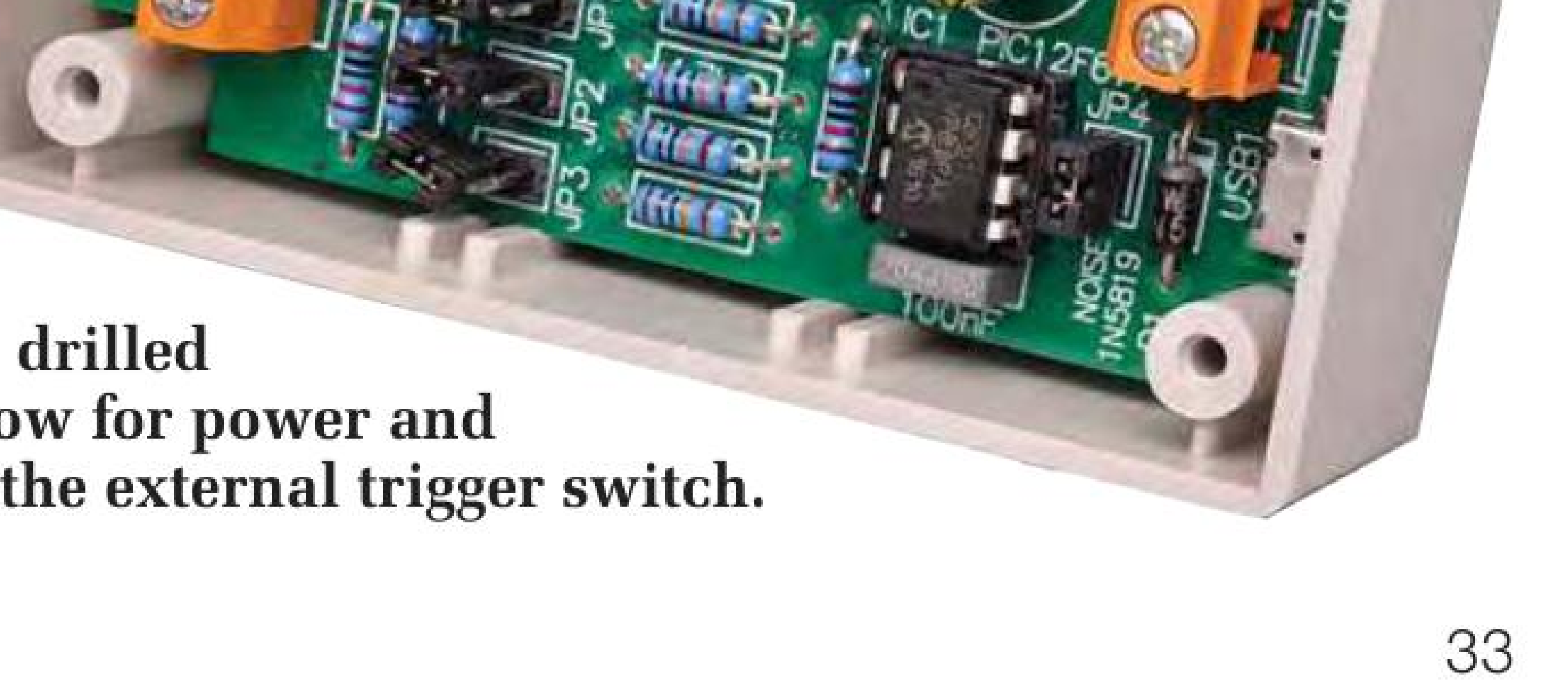
Testing

With IC1 and IC2 out of their sockets, apply power either via a 5V DC supply connected to CON1, or using a USB cable from a computer or USB supply.

Check that LED1 lights up and that there is around 5V between pins 1 and 8 of the sockets for IC1 and IC2. If power is applied to CON1, the reading will probably be closer to 4.7V due to diode D1.

Regardless, you should get a reading between 4.5V and 5.25V. You can then remove power from the circuit.

If you have purchased pre-programmed microcontrollers then you



Assuming you're building this for a model train layout, the PCB can be mounted 'as-is' wherever there is a suitable place on or in your layout. But it is also designed to clip into the side rails of a UB3 Jiffy box, as shown here, with the LED just poking through the front panel. Naturally, access holes will need to be drilled in the ends of the box to allow for power and speaker wiring, along with the external trigger switch.

Changing the whistle or horn sound

Changing the whistle period

With no jumpers inserted for JP1-JP3, rotate VR1 to adjust the whistle period, up to a maximum of 22.5s (fully clockwise).

If you are not using the simulated Doppler effect, the whistle period can be extended indefinitely by holding down S1.

Changing the volume rise rate and Doppler effect

This applies to the steam whistle sound only (ie, not the diesel horn). Switch off power and insert jumper shunts for JP1, JP2 and JP3. Power up and set VR1 for the desired ramp time, with a more clockwise position selecting a longer ramp.

To disable the Doppler effect, set VR1 to a position between fully anti-clockwise and halfway.

With VR1 fully anti-clockwise, the initial volume ramp-up is almost instant, while if you set it just slightly less than its midpoint, you will get a ramp-up time of around eight seconds. Intermediate settings give a shorter ramp-up time.

To enable the Doppler effect, set VR1 to a position between halfway and fully clockwise. The ramp setting is similar, ie, just above halfway will give you an almost instantaneous ramp-up while setting VR1 fully clockwise sets the ramp-up time to around eight seconds.

In this mode, there is also a volume ramp-down period at the end of the effect and it is the same time as the ramp-up.

When VR1 is set at the required position, press S1 so that the setting is stored in Flash memory.

Switch off power and remove the three jumper shunts. Set VR1 back to the required position for the whistle period when you've finished. You can then power it back up and press S1 to test the new setting.

Note that it's generally a good idea to set the whistle period in Doppler Mode to be slightly longer than twice the ramp-up/ramp-down time. This way, the volume will rise to maximum and then almost immediately begin to fall again, as if the train has just passed.

You may need to tweak the volume rate and whistle period a few times to get the desired effect.

Changing the oscillator frequencies

The table below shows some suggested sets of oscillator frequencies to simulate a steam whistle and various diesel horns, but note that you are not restricted to just using these frequencies.

The first entry shown gives the frequencies that the unit will default to the first time it is powered up.

The oscillator frequencies are changed one at a time. Start by switching off power and then place two jumper shunts on JP1-JP3, leaving out the jumper in the channel that you want to adjust. Remove JP4 to disable steam noise for the moment.

Now attach a frequency meter (eg, a DMM with frequency measurement) to the right-most pin on the unused jumper header.

Connect its ground reference to the ground PC stake (between S1 and CON2).

Apply power and adjust VR1 until your meter reads the frequency required. It can be adjusted over the range of 244-1053Hz. The frequency will change in steps of 3Hz at the low end, rising to 48Hz at the high end.

You will be able to hear the oscillator if the speaker is connected. When you have settled on the required frequency, press S1 to store the value in Flash memory.

To adjust another oscillator, disconnect power and move one of the jumper shunts, then re-apply power and go through the same procedure again.

When you have finished, switch off and remove JP1-JP3. Re-install JP4 if you removed it earlier.

Set VR1 back to the required position for the whistle period now that you have finished adjusting the oscillator frequencies.

Sound effect	Oscillator 1	Oscillator 2	Oscillator 3	White noise
Steam whistle (default)	740Hz	525Hz	420Hz	Yes (JP4 in)
2-Car Diesel	600Hz	520Hz	420Hz	No (JP4 out)
40-43, 4401-4440 Diesel	277Hz	329Hz	440Hz	No (JP4 out)
422, 442, 73, 48126 Diesel	548Hz	322Hz	429Hz	No (JP4 out)

Suggested oscillator frequencies for various whistles or horns.

The Doppler effect

The Doppler effect is a phenomenon that occurs when an object is moving towards you. Any sound that it generates will appear to be higher in pitch than usual. And similarly, if that object is moving away from you, the sound will appear to be lower in pitch. For example, it is very obvious when a vehicle with a siren passes you at high speed.

It happens because sound waves travel through the air at approximately 340m/s (1200km/h) and as the speed of the generating object (relative to you) becomes a significant fraction of that, the sound waves pass you at a noticeably different rate, thus altering the perceived pitch.

To find out more about the Doppler effect, see: **https://en.wikipedia.org/wiki/Doppler_effect**

Since trains can travel quite fast, the Doppler effect can be very apparent, especially when they use their whistle or horn as they are passing you. At 80km/h, the pitch changes by about 6.5% in each direction and the overall 13% difference is very noticeable. So we have included a facility to simulate this.

can plug IC1 and IC2 into their sockets now, making sure that they are oriented correctly and that the correct programmed IC is in the correct socket. The chip programmed as the noise generator is IC1.

If your chips have not already been programmed, you will need to program them first, using HEX files which can be downloaded from the March 2020 page of the *PE* website. (Do note that there are two different files for IC1 and IC2.)

Connect the speaker to CON1 and apply power again. Set VR1 fully anticlockwise and press and hold S1. You should be greeted by the steam train whistle sound.

If you don't hear the steam noise in the background, check that JP4 has been inserted.

Installation

The board is too large to mount inside a locomotive, so if you want to use it as part of a model railway layout, the best place to put it would be underneath the layout; for example, near a station.

You could then mount the speaker inside the station. It could be triggered manually or via a reed switch or microswitch as the train passes.

Whatever kind of switch you are using, connect it between pins 1 and 2 of CON2.

Or if you are using a microcontroller to trigger it, connect a digital output pin on that micro to pin 3 of CON2 with the micro's ground going to pin 2. You can apply anywhere between 1.5V and 12V to this pin to trigger the unit.

Reproduced by arrangement with
SILICON CHIP magazine 2020.
www.siliconchip.com.au

STORE YOUR BACK ISSUES ON CD-ROMS

VOL 1	**BACK ISSUES**	– January 1999 to June 1999 Plus bonus material from Nov and Dec 1998
VOL 2	**BACK ISSUES**	– July 1999 to December 1999
VOL 3	**BACK ISSUES**	– January 2000 to June 2000
VOL 4	**BACK ISSUES**	– July 2000 to December 2000
VOL 5	**BACK ISSUES**	– January 2001 to June 2001
VOL 6	**BACK ISSUES**	– July 2001 to December 2001
VOL 7	**BACK ISSUES**	– January 2002 to June 2002
VOL 8	**BACK ISSUES**	– July 2002 to December 2002
VOL 9	**BACK ISSUES**	– January 2003 to June 2003
VOL 10	**BACK ISSUES**	– July 2003 to December 2003
VOL 11	**BACK ISSUES**	– January 2004 to June 2004
VOL 12	**BACK ISSUES**	– July 2004 to December 2004
VOL 13	**BACK ISSUES**	– January 2005 to June 2005
VOL 14	**BACK ISSUES**	– July 2005 to December 2005
VOL 15	**BACK ISSUES**	– January 2006 to June 2006
VOL 16	**BACK ISSUES**	– July 2006 to December 2006
VOL 17	**BACK ISSUES**	– January 2007 to June 2007
VOL 18	**BACK ISSUES**	– July 2007 to December 2007
VOL 19	**BACK ISSUES**	– January 2008 to June 2008
VOL 20	**BACK ISSUES**	– July 2008 to December 2008
VOL 21	**BACK ISSUES**	– January 2009 to June 2009
VOL 22	**BACK ISSUES**	– July 2009 to December 2009
VOL 23	**BACK ISSUES**	– January 2010 to June 2010

ONLY £16.45 each
Five years just £29.95
including VAT and p&p

VOL 24	**BACK ISSUES**	– July 2010 to December 2010
VOL 25	**BACK ISSUES**	– January 2011 to June 2011
VOL 26	**BACK ISSUES**	– July 2011 to December 2011
VOL 27	**BACK ISSUES**	– January 2012 to June 2012
VOL 28	**BACK ISSUES**	– July 2012 to December 2012
VOL 29	**BACK ISSUES**	– January 2013 to June 2013
VOL 30	**BACK ISSUES**	– July 2013 to December 2013
VOL 31	**BACK ISSUES**	– January 2014 to June 2014
VOL 32	**BACK ISSUES**	– July 2014 to December 2014
VOL 33	**BACK ISSUES**	– January 2015 to June 2015
VOL 34	**BACK ISSUES**	– July 2015 to December 2015
VOL 35	**BACK ISSUES**	– January 2016 to June 2016
VOL 36	**BACK ISSUES**	– July 2016 to December 2016
VOL 37	**BACK ISSUES**	– January 2017 to June 2017
VOL 38	**BACK ISSUES**	– July 2017 to December 2017
VOL 39	**BACK ISSUES**	– January 2018 to June 2018
VOL 40	**BACK ISSUES**	– July 2018 to December 2018
VOL 41	**BACK ISSUES**	– January 2019 to June 2019

FIVE-YEAR DVD/CD-ROMs

No.1 – Jan 03 to Dec 07
No.2 – Jan 04 to Dec 08
No.3 – Jan 05 to Dec 09
No.4 – Jan 06 to Dec 10
No.5 – Jan 07 to Dec 11
No.6 – Jan 08 to Dec 12
No.7 – Jan 09 to Dec 13
No.8 – Jan 10 to Dec 14
No.9 – Jan 11 to Dec 15
No.10 – Jan 12 to Dec 16
No.11 – Jan 13 to Dec 17
No.12 – Jan 14 to Dec 18 – NEW!

15 YEARS OF EPE!

AMAZING BUNDLE OFFER

No.1 – Jan 03 to Dec 07
PLUS
No.6 – Jan 08 to Dec 12
PLUS
No.11 – Jan 13 to Dec 17

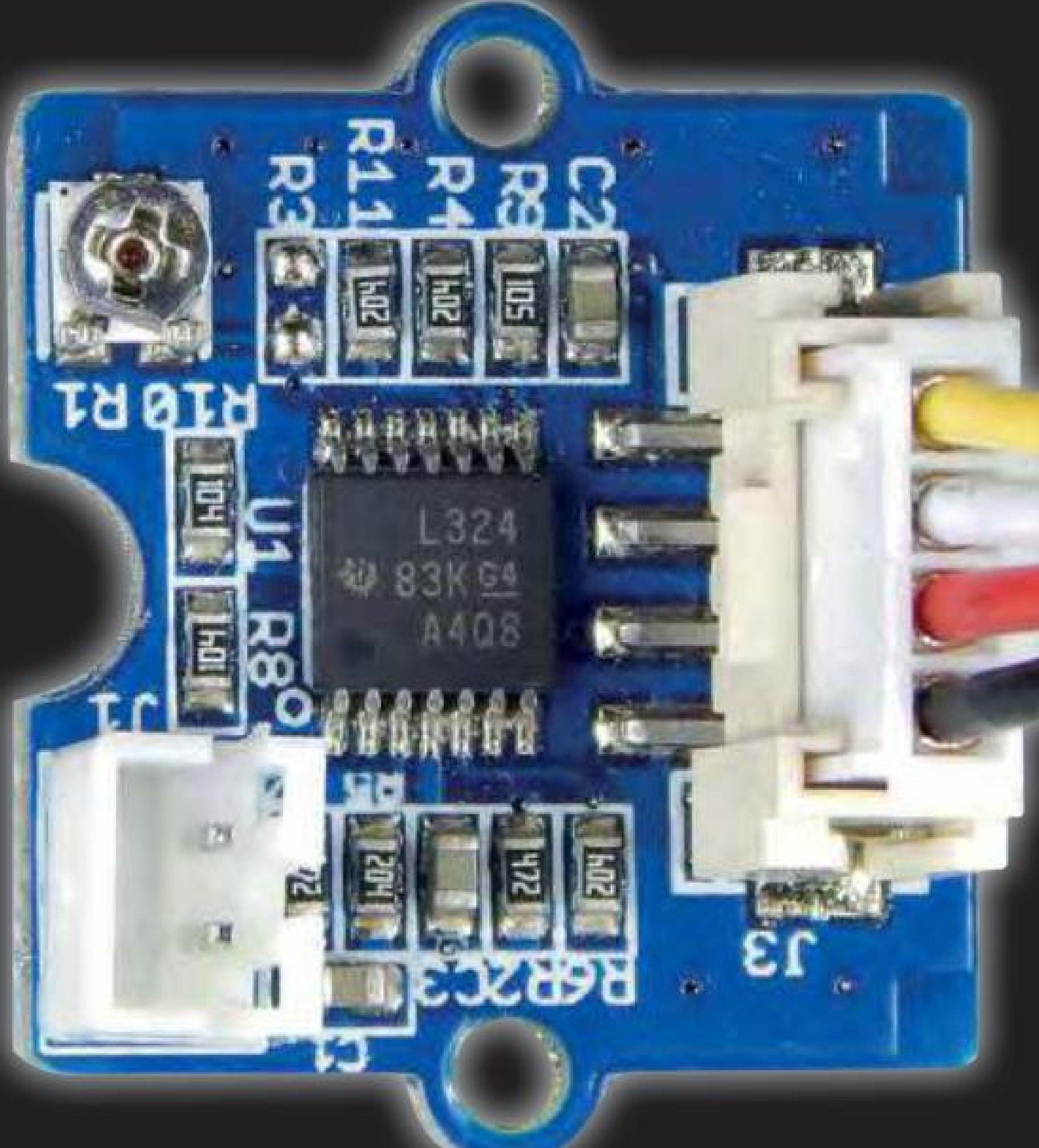
ALL FOR JUST
£39.95

All DVD/CD-ROMs are suitable for any PC with a DVD/CD-ROM drive.
They require Adobe Acrobat Reader (free from: www.adobe.com/acrobat)

ORDER YOUR CD-ROMs TODAY!
JUST CALL 01202 880299 – OR VISIT www.epemag.com

Using Cheap Asian Electronic Modules Part 22: by Jim Rowe

Galvanic Skin Response



This Seeed/Grove-designed galvanic skin response sensor measures changes in the resistance of human skin, which indicate changes in mood, apprehension or other psychological phenomena. It's smaller than a stamp and comes with a pair of sensing electrodes. It also has an analogue voltage output, making it easy to use with any micro or a digital multimeter.

These days, the phrase 'galvanic skin response' (GSR) is regarded as obsolete; instead, it is known as 'electrodermal activity'. or EDA. Nonetheless, GSR is still pretty widely used.

GSR is often regarded as the primary body parameter measured in 'lie detectors', or 'polygraphs' as they're known in the US. However, GSR is only one of the many physiological indicators monitored in polygraphs; others are blood pressure, pulse rate and respiration.

We should point out that despite the widespread use of polygraphs throughout the US and other countries, there is a great deal of doubt in scientific circles about their accuracy and reliability. They supposedly can indicate when a person gives false answers to questions. However, polygraph evidence is mostly inadmissible in UK courts and most other countries.

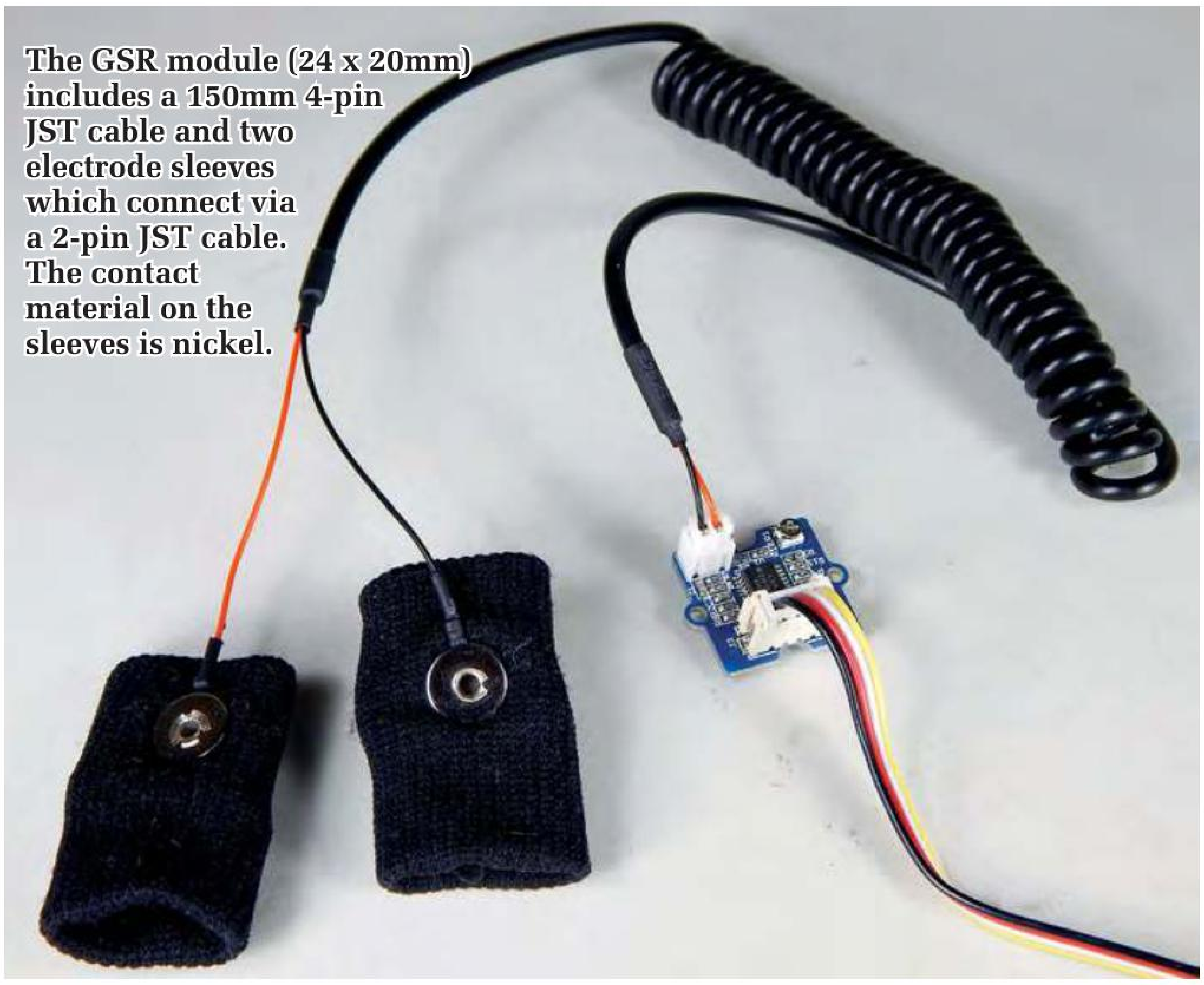
The first suggestion that human sweat glands were involved in creating changes in the electrical conductivity of the skin was made in Switzerland in 1878, by researchers Hermann and Luchsinger. Then in 1888, the French neurologist Fere demonstrated that skin conductivity could be changed by emotional stimulation and also that this could be inhibited by drugs.

Pioneering psychoanalyst Carl Jung, in his book *Studies in Word Analysis* (1906), described experiments using a GSR meter to evaluate the emotional sensitivities of patients to lists of words during word association sessions.

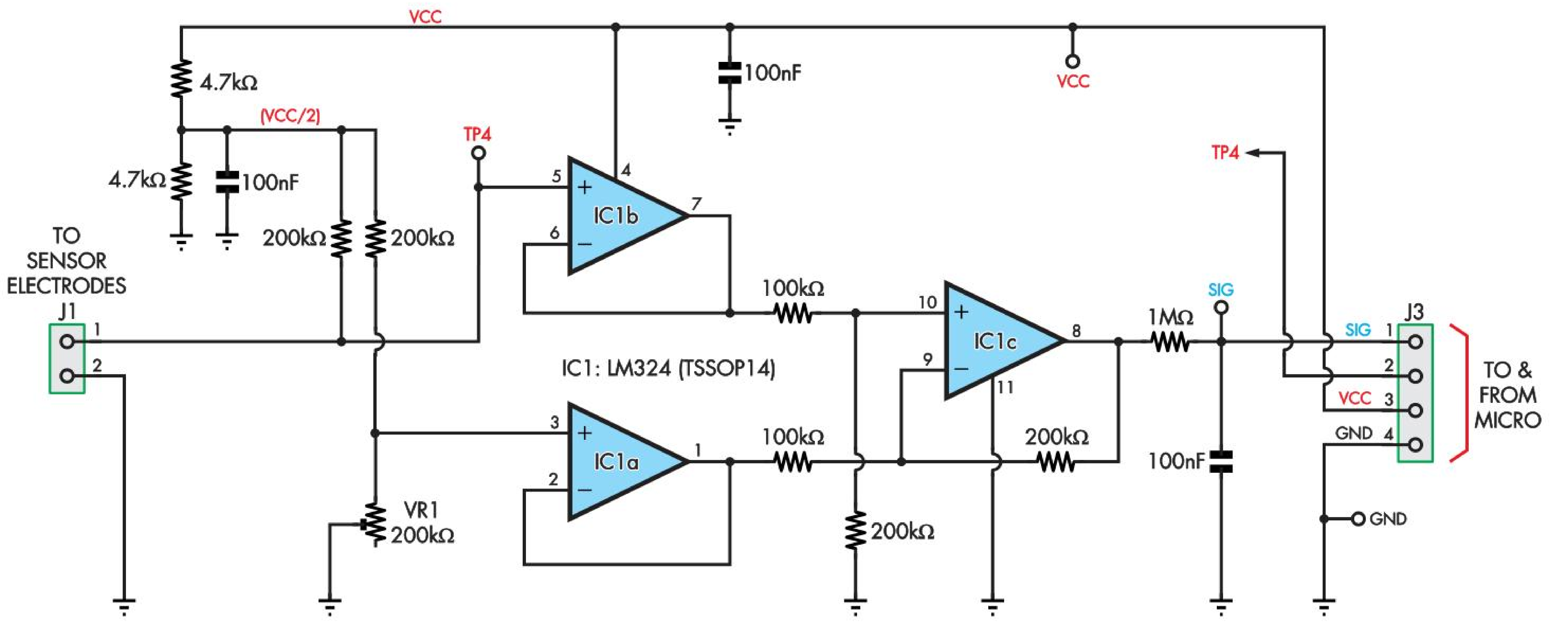
Although the first polygraph was invented in 1921 by John Augustus Larson at the University of California, it only monitored blood pressure and respiration. Larson's protege Leonarde Keeler updated the device in 1939 by making it portable and adding the monitoring of GSR. His device was purchased by the FBI and then became the prototype of the modern polygraph.

So what is GSR/EDA?

The electrical conductivity of our skin is not under conscious control, but modulated by our sympathetic autonomous (subconscious) nervous system. Therefore, it responds to our cognitive and emotional states.



The GSR module (24 x 20mm) includes a 150mm 4-pin JST cable and two electrode sleeves which connect via a 2-pin JST cable. The contact material on the sleeves is nickel.



Seeed/Grove GSR Sensor

Fig.1: complete circuit diagram for the Seeed/Grove GSR sensor module. Non-inverting input pin 5 of IC1 varies from 0-2.5V (5V DC supply) depending on the conductivity of your skin. VR1 adjusts the voltage at pin 3 of IC1a. The difference between these appears at the pin 8 output of IC1c and goes through a low-pass filter, and then onto pin 1 of J3.

Initially, it was thought that modulation of sweat gland activity by the sympathetic nervous system was solely responsible for the changes in GSR/EDA, and this is still regarded as the main factor. However, it's now believed that there are also accompanying changes in blood flow and muscular activity which affect conductivity.

GSR/EDA sensors are usually fitted to the fingers because our hands and feet have the highest density of sweat glands on our bodies (200-600 sweat glands per cm^2). In fact, the palms of our hands and the inside of our fingers are ideal locations for sensing GSR/EDA, and you don't have to take off your shoes and socks!

The Seeed/Grove GSR module

The Seeed/Grove-designed GSR sensing module is tiny, measuring only 24 × 20 × 9mm, including the two JST 2.0 PH-series SIL headers.

The unusual shape of the PCB, with semicircular cut-outs at two ends which host the 2mm mounting holes, is because the module was designed as part of Seeed Studio's 'Grove' module system, a standardised prototyping system.

There are many modules available in the Grove system, including sensors for light, IR, temperature, gas, dust, acceleration and the Earth's magnetic field – to name just a few.

All of these modules have a standardised connector system, and Seeed has also produced shields and similar 'piggyback' boards to make it easy to connect multiple Grove modules to micros like the Arduino, the Raspberry Pi and the Beaglebone series.

Since the modules come with a cable fitted with a 4-pin JST 2.0 connector at each end, it's quite easy to connect a single module like this to a board such as a Micromite, or even to a digital multimeter (DMM).

This module isn't quite as affordable as some of the other modules we've looked at in these articles, perhaps because it comes with a pair of 'finger sock' electrode sleeves, together with suitable cables to connect to the module. It also comes with the aforementioned 150mm-long cable for connection to the micro.

The cost for the module plus these extra parts is around US$16, and is available from the usual web outlets (GearBest, Amazon and eBay). There's also a very similar module made by SichiRay, available from AliExpress.

Inside the module

There's not a great deal to the Seeed/Grove GSR sensor module, as you can see from Fig.1. It uses an SMD version of the LM324 quad op amp (IC1), with three of its amplifiers connected in the standard instrumentation amplifier configuration. IC1c is used as a standard differential amplifier with a gain of 2.0, while IC1b and IC1a are unity-gain buffers driving its two inputs.

But instead of having a gain-setting resistor connected between the inverting inputs (–) of IC1b and IC1a, as is typically the case with a purpose-designed instrumentation amplifier, the input buffers are left with unity gain.

To the left of IC1b and IC1a is the simple circuitry used to sense the skin conductivity between the two sensing electrodes, which are connected to J1. At the top is a resistive voltage divider which derives a reference voltage of $V_{CC} \div 2$, or 2.5V when the module is powered from a 5V supply.

This reference voltage is used to bias non-inverting (+) inputs of both IC1b and IC1a via 200kΩ series resistors.

Since pin 1 of J1 is connected to the + input of IC1b (pin 5), the voltage at this pin will vary according to the skin conductivity between the two electrodes. On the other hand, the + input of IC1a (pin 3) is simply connected via small trimpot VR1 to ground, and the pin 2 input of J1 also connects to ground.

Fig.2: the GSR sensor can be easily tested by powering it via a USB supply (eg, a computer) for the required 5V DC and connecting the analogue voltage output to a DMM.

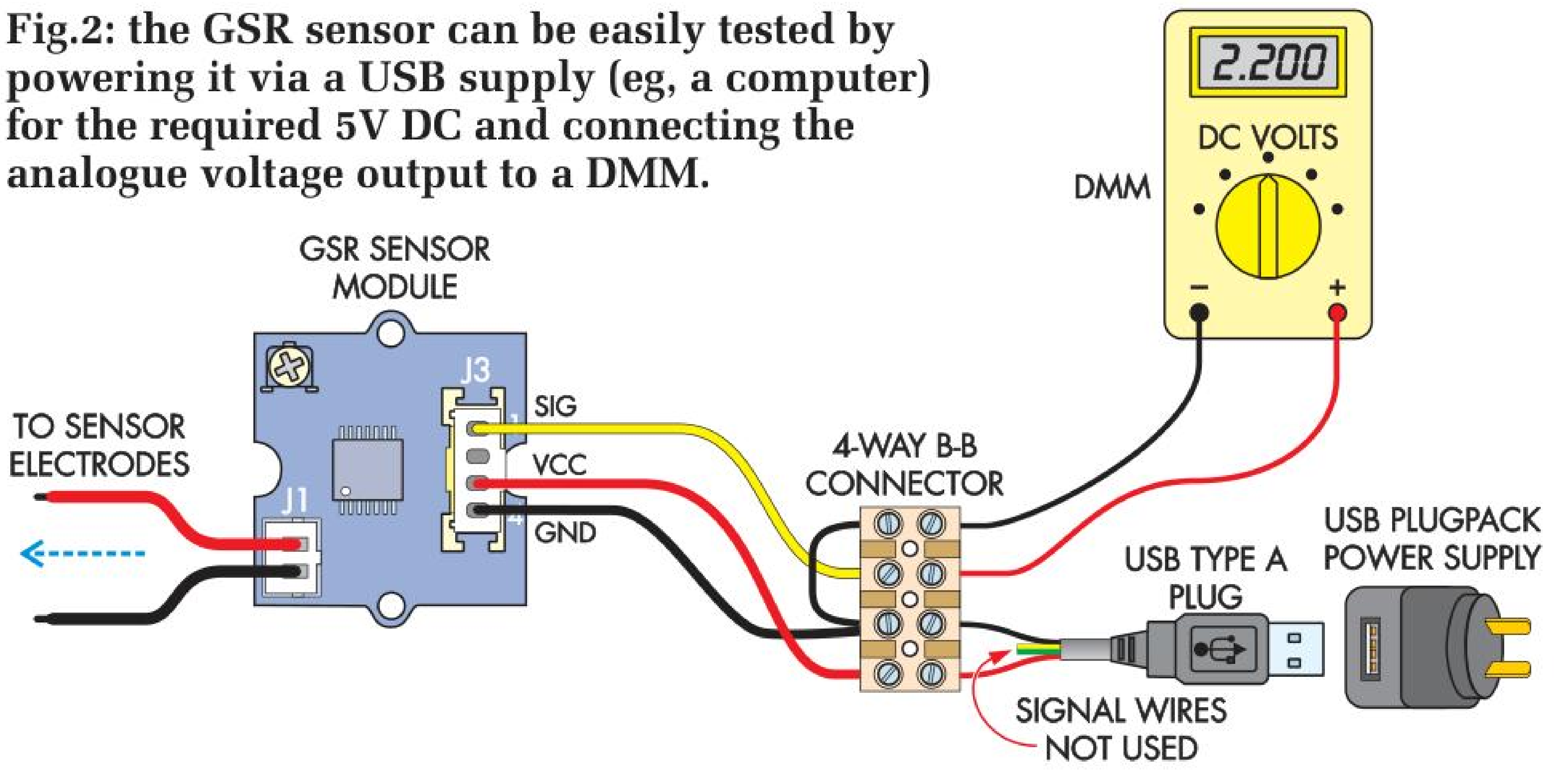
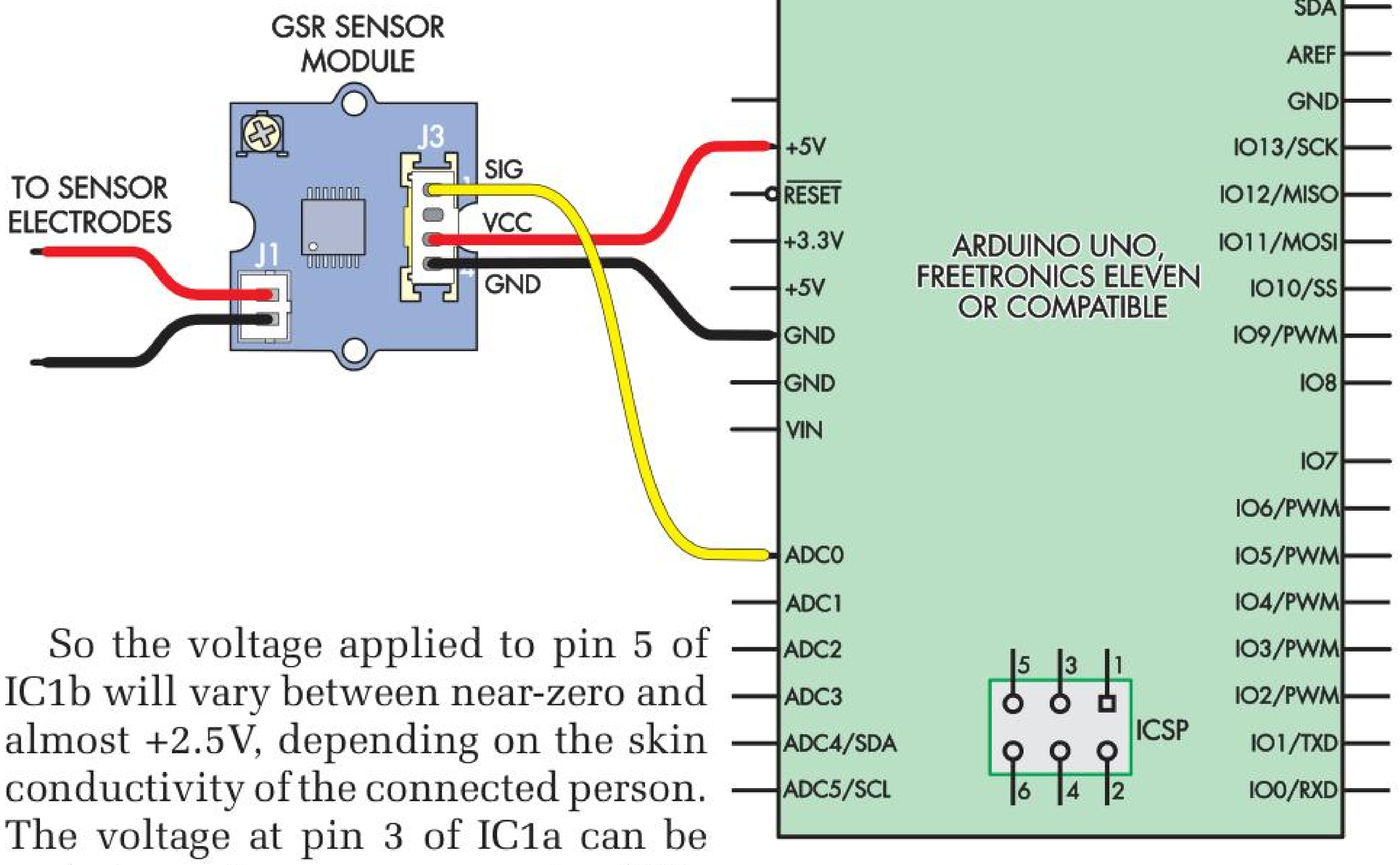


Fig.3: wiring diagram for the GSR module to an Arduino module. Output pin SIG must be connected to an analogue input pin.


GSR SENSOR MODULE
TO SENSOR ELECTRODES
J1
J3
SIG
VCC
GND
DC VOLTS INPUT
USB TYPE B MICRO
ARDUINO UNO, FREETRONICS ELEVEN OR COMPATIBLE
+5V
RESET
+3.3V
+5V
GND
GND
VIN
ADC0
ADC1
ADC2
ADC3
ADC4/SDA
ADC5/SCL
SCL
SDA
AREF
GND
IO13/SCK
IO12/MISO
IO11/MOSI
IO10/SS
IO9/PWM
IO8
IO7
IO6/PWM
IO5/PWM
IO4/PWM
IO3/PWM
IO2/PWM
IO1/TXD
IO0/RXD
ICSP
5
3
1
6
4
2

So the voltage applied to pin 5 of IC1b will vary between near-zero and almost +2.5V, depending on the skin conductivity of the connected person. The voltage at pin 3 of IC1a can be varied over the same range using VR1. This allows VR1 to set the full-scale output voltage of the module when the electrodes are open-circuit.

Note that when the electrodes are worn, the maximum current that could flow between them is 12.5µA (2.5V ÷ 200kΩ). This is too low to be consciously sensed, and certainly not enough to give an electric shock.

So the variations in skin conductivity between the two sensing electrodes connected to J1 cause changes in the voltage difference between pins 5 and 3 of IC1.

The output voltage from pin 8 of IC1c is this difference. A simple 2Hz low-pass filter comprising a 1MΩ series resistor and a 100nF capacitor is connected between pin 8 of IC1c and pin 1 of J3, which is the power supply/output connector.

Pin 2 of J3 is connected to TP4 and pin 5 of IC1b, which allows you to monitor the voltage across the GSR electrodes with a DMM if necessary.

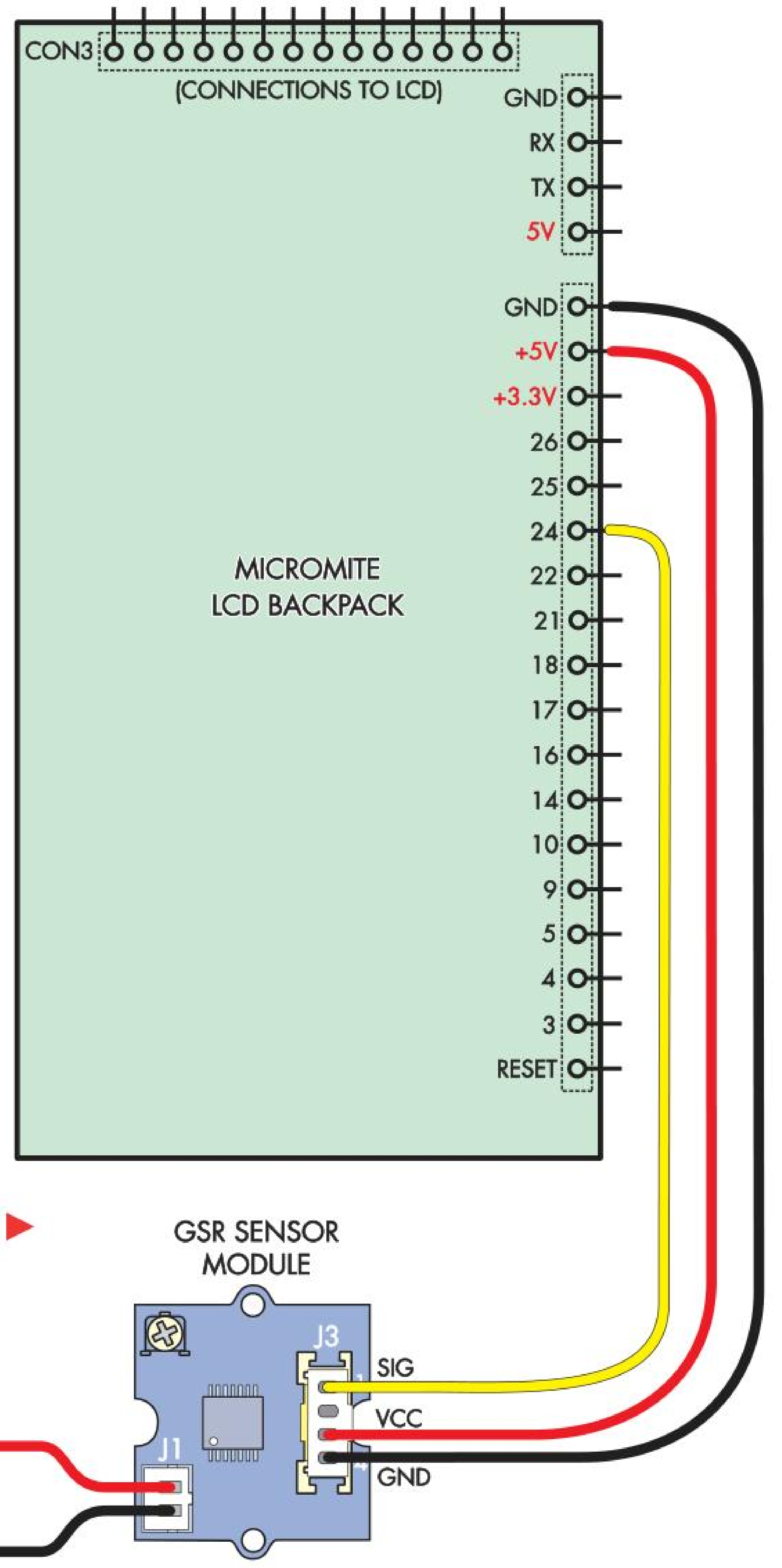

CON3
(CONNECTIONS TO LCD)
GND
RX
TX
5V
GND
+5V
+3.3V
26
25
24
22
21
18
17
16
14
10
9
5
4
3
RESET
MICROMITE LCD BACKPACK
GSR SENSOR MODULE
TO SENSOR ELECTRODES
J1
J3
SIG
VCC
GND

Fig.4: wiring diagram for the GSR module to a Micromite BackPack.

Trying it out

Probably the simplest way of trying out this module is to provide it with a source of 5V DC and use a DMM to monitor its analogue output voltage, as shown in Fig.2. The 5V power supply for the module can come from virtually any USB supply, since it only draws about 1.2mA.

Fig.3 shows how the Seeed/Grove GSR module can be connected to an Arduino Uno or an equivalent microcontroller board, while Fig.4 shows how it's connected to a Micromite LCD BackPack (see the article in the May 2017 issue of *PE*).

In both cases, the V_{CC} and GND pins of the module's output connector (J3) are connected to +5V and GND respectively, while the SIG output pin is connected to the A0 pin of the Arduino, or to pin 24 of the Micromite.

I found a very simple sketch for the Arduino in one of Seeedstudio's wikis (**http://wiki.seeedstudio.com/Grove-GSR_Sensor/**). It merely makes a series of 10 measurements of the module's output voltage, adds them together and then divides by 10 to get their average.

This is then sent back to your PC, to be either printed out in Serial Monitor or plotted using Serial Plotter. Then it loops back and repeats this sequence over and over again.

You can see a sample output plot from this sketch in Fig.6. It's called **GSR_Testing_sketch.ino** and we've made it available as a download from the March 2020 page of the *PE* website.

Note that when you first power up the Arduino with the module connected, it's a good idea to set trimpot VR1 to give a readout of around 512 before the electrodes are fitted to anyone's fingers. This only needs to be done once, not every time you apply the power.

For those who want to use the GSR module with a Micromite, I have written a small program in MMBasic. This is identical to the Arduino program, taking a series of 10 measurements and calculating their average. The measurements are then sent back to the PC for display in the MMChat window. It's also shown on the Micromite's LCD screen as a single figure, which changes with each new set of measurements.

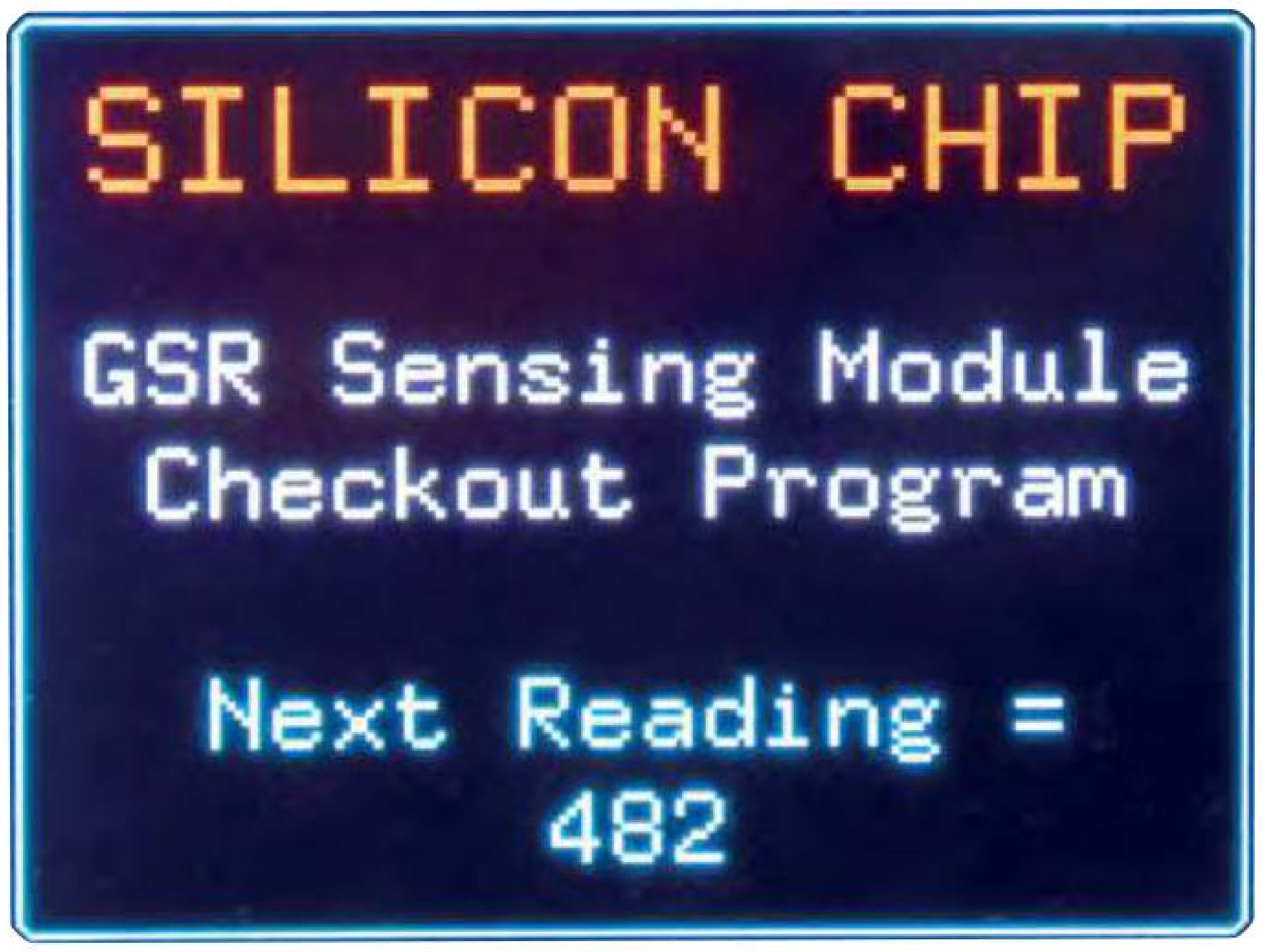
Fig.5 shows a screen grab of this program in operation. It's called **GSR module checkout.bas** and is also available as a download from the March 2020 page of the *PE* website.

This should provide you with a starting place for writing a more elaborate program of your own, perhaps one that displays the growing GSR plot on your PC's screen, like a polygraph display. Once again, it's a good idea to adjust VR1 for a reading of around 512 before the electrodes are fitted.

Breadboarding it

Given how simple the circuit shown in Fig.1 is, you may be wondering whether it's possible to breadboard it. We reckon it wouldn't be too hard.

The only thing you need to be careful of is to avoid any possible leakage currents on the tracks and components connected to the non-inverting inputs of IC1a and IC1b (pins 3 and 5), as this could disturb the readings, especially if the leakage currents were to vary with temperature, humidity etc.



```
SILICON CHIP

GSR Sensing Module
 Checkout Program

 Next Reading =
      482
```

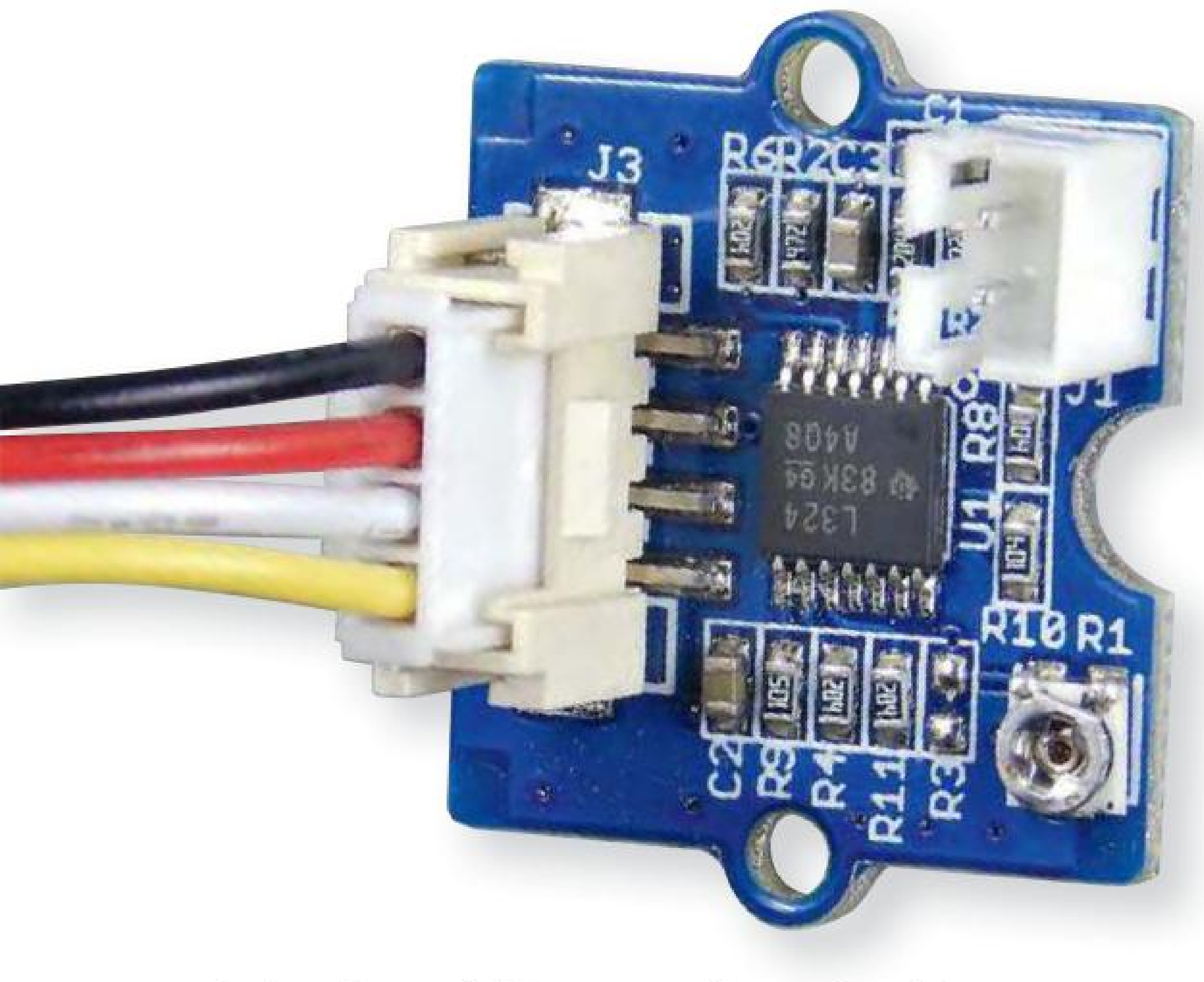
Fig.5: the sample program running on a Micromite. Connect two fingers to the sensors to display the current skin resistance. Anything ±5% from those initial values indicates a change in mood. A higher reading typically indicates a more relaxed mood, while a lower reading is a tenser mood (greater perspiration, thus decreasing skin resistance).

This generally means keeping the breadboard and components plugged into it clean and dry and avoid touching it during operation.

You could probably even build a little GSR module yourself on a bit of veroboard, using a DIP LM324 IC and a handful of passives, in a similar arrangement to that shown in Fig.1.

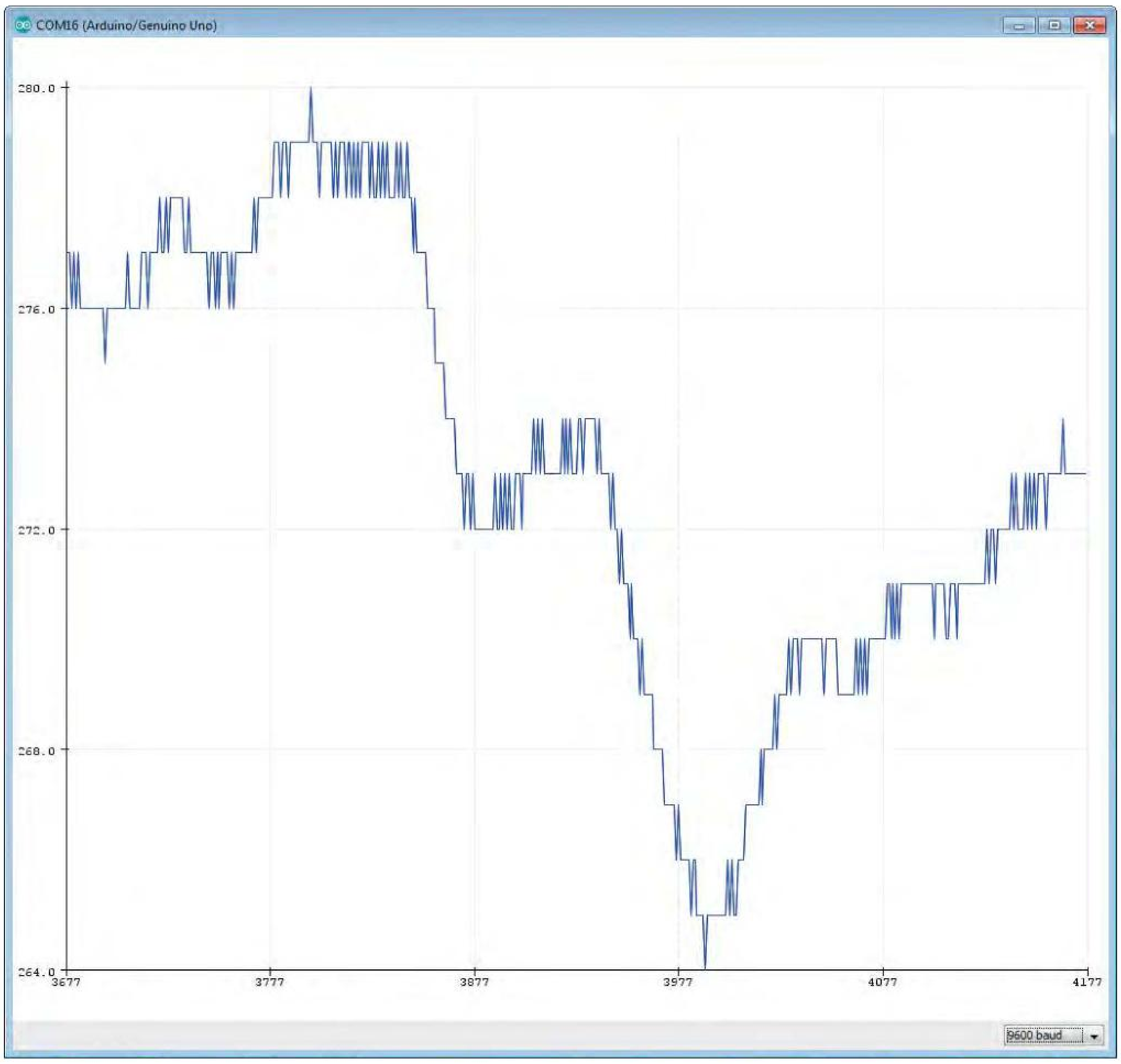
Useful links

http://bit.ly/pe-mar20-eda
http://bit.ly/pe-mar20-poly
http://bit.ly/pe-mar20-grove



The Seeed/Grove galvanic skin response module, shown above at twice actual size, is based on an LM324 quad op amp and costs around US$16.00.

Reproduced by arrangement with
SILICON CHIP magazine 2020.
www.siliconchip.com.au



Output plot of the values from the GSR module using the Arduino Serial plotter. The values swing from a high of 280 to a low of 264, even though the reference value is 512, due to the way the module is designed.

PoLabs

For more information or software download please visit

www.poscope.com/epe

PoKeys
Connect, Control

- USB
- Ethernet
- Web server
- Modbus
- CNC (Mach3/4)
- IO
- PWM
- Encoders
- LCD
- Analog inputs
- Compact PLC

Stepper motor drivers
Drive

- up to 256 microsteps
- 50 V / 6 A
- USB configuration
- Isolated
- up to 32 microsteps
- 30 V / 2.5 A

PoScope Mega1+
PoScope Mega50
Measure

- up to 50MS/s
- resolution up to 12bit
- Lowest power consumption
- Smallest and lightest
- 7 in 1: Oscilloscope, FFT, X/Y, Recorder, Logic Analyzer, Protocol decoder, Signal generator

PIC n' Mix

Mike Hibbett's column for PIC project enlightenment and related topics

Audio Spectrum Analyser – design update

We change tack this month and return to our series covering audio spectrum analysis. In the original series of article on this subject last year we developed a circuit design for a low-frequency audio spectrum analyser covering 100Hz – 1kHz. Evolving from a design for a simple guitar tuner, this grew into to a fully fledged audio spectrum display, using a 2.5-inch colour LCD and a dsPIC33 processor, covering the low end of the audio range up to 1kHz.

That initial design had some issues (as all initial designs do) so we return now to correct those errors and expand on the functionality.

The first issue to address is that the original PCB design, as shown in Fig. 1, required some wire straps and two PCB traces to be cut. These corrected connection errors to the processor's internal op amp. These issues will be easy to correct in a second PCB iteration. (The first physical realisation of any PCB design is rarely perfect – in our experience, never!)

PSU noise

In addition to correcting the bugs in our PCB design, months of experimentation and use of this circuit have brought out a number of desired design changes to the hardware and user interface. The most important issue we noticed was that the use of a DC-DC converter as a power supply from batteries increased the noise on the audio input signal. This is not terribly surprising; a simple battery, as used on our original design introduces no noise, so any switching circuit used in its place will have a negative impact. As engineers, our job is to minimise that impact.

PSU options

Thinking about a 'revision 2' of this board we have decided to provide a second option for the power supply. You can choose a low-efficiency / low-noise LDO (low drop-out) power supply driven by four alkaline batteries, or an efficient DC-DC converter power supply that can run from one, two or three alkaline batteries, as used now. We will add both circuit options to the PCB design so you can decide which to use.

We also want to extend the original input range of 0 – 1kHz to the entire hearing (audio) range, up to 20kHz. That makes it suitable for younger readers' ears – beyond the age of 20 hearing range starts to decrease (and at 55 we are happy to be able to hear up to 12kHz!).

In this series of articles we will also create a 3D-printable enclosure for the final product. This will be a first for *Practical Electronics*, but since we introduced hobbyist 3D printers way back in 2009 and ten years later we find they cost as little as £130, it's reasonable to expect that these devices are becoming a feature in hobbyist workshops. (In fact, to keep abreast of the times, we will be discussing the introduction of hobbyist laser cutters in an article later this year!)

So, in summary, the changes we are going to make are:

- Correct the PCB traces for the op amp connections
- Add an optional low-noise LDO DC supply
- Increase the audio range to 20kHz
- Provide additional user interface features
- Create a 3D printable enclosure.

The second-to-last feature in that list is the only one that involves software changes to the original design. Let's talk through those changes first, as they will be the most challenging to implement in the coming months.

User interface

Currently, the LCD display shows no textual indication of frequency – it's just a constantly varying signal ranging from zero to 1kHz. This is fun to play with but not terribly practical. In this update we will provide options to:

- Display the value of the peak frequency
- Allow a button press to choose the frequency span
- Allow a button press to toggle peak trace hold.

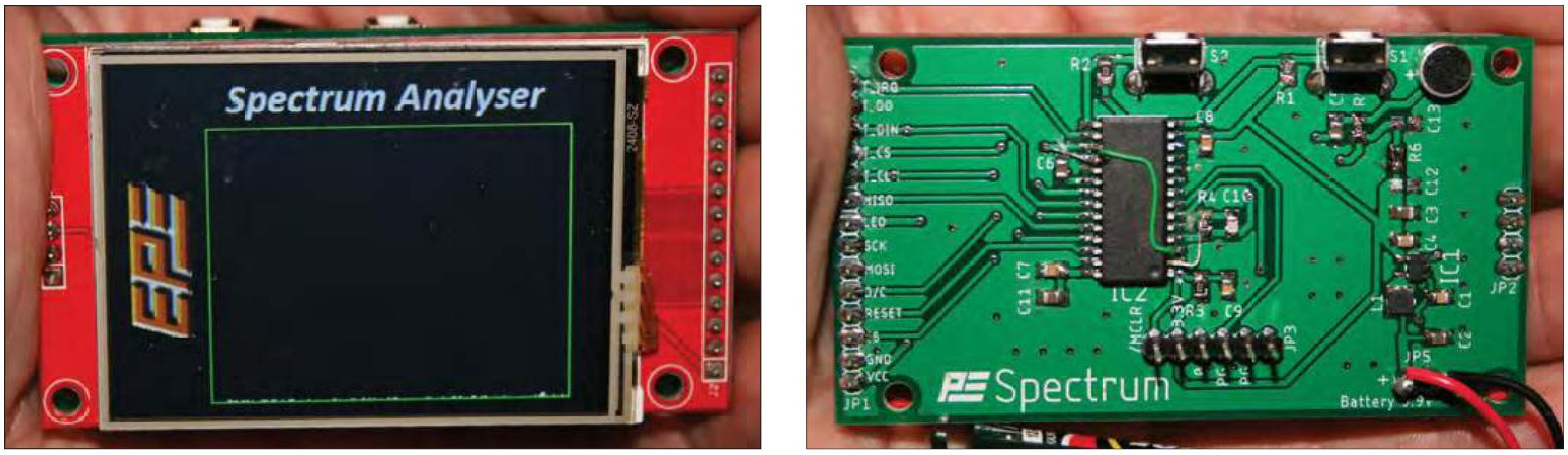


Fig.1. Original Spectrum Analyser design with PCB modifications.

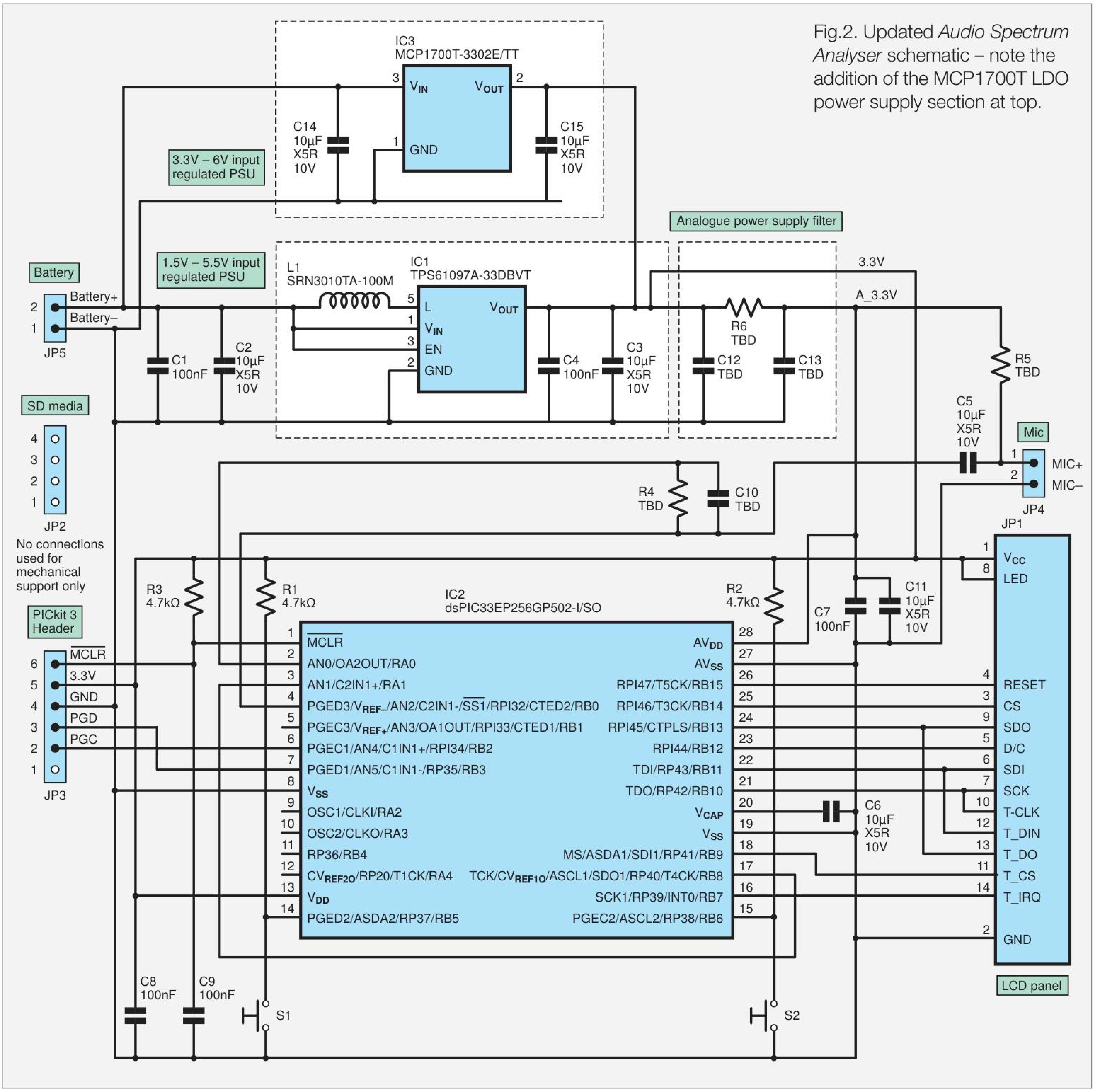

IC3
MCP1700T-3302E/TT
VIN
VOUT
GND
C14
10µF
X5R
10V
C15
10µF
X5R
10V
3.3V – 6V input regulated PSU
Battery
Battery+
Battery–
JP5
1.5V – 5.5V input regulated PSU
L1
SRN3010TA-100M
IC1
TPS61097A-33DBVT
L
VIN
EN
GND
VOUT
C1
100nF
C2
10µF
X5R
10V
C4
100nF
C3
10µF
X5R
10V
Analogue power supply filter
R6
TBD
C12
TBD
C13
TBD
3.3V
A_3.3V
R5
TBD
C5
10µF
X5R
10V
Mic
MIC+
MIC–
JP4
SD media
JP2
No connections used for mechanical support only
R4
TBD
C10
TBD
JP1
PICkit 3 Header
MCLR
3.3V
GND
PGD
PGC
JP3
R3
4.7kΩ
R1
4.7kΩ
IC2
dsPIC33EP256GP502-I/SO
R2
4.7kΩ
C7
100nF
C11
10µF
X5R
10V
MCLR
AN0/OA2OUT/RA0
AN1/C2IN1+/RA1
PGED3/VREF–/AN2/C2IN1-/SS1/RPI32/CTED2/RB0
PGEC3/VREF+/AN3/OA1OUT/RPI33/CTED1/RB1
PGEC1/AN4/C1IN1+/RPI34/RB2
PGED1/AN5/C1IN1-/RP35/RB3
VSS
OSC1/CLKI/RA2
OSC2/CLKO/RA3
RP36/RB4
CVREF2O/RP20/T1CK/RA4
VDD
PGED2/ASDA2/RP37/RB5
AVDD
AVSS
RPI47/T5CK/RB15
RPI46/T3CK/RB14
RPI45/CTPLS/RB13
RPI44/RB12
TDI/RP43/RB11
TDO/RP42/RB10
VCAP
VSS
MS/ASDA1/SDI1/RP41/RB9
TCK/CVREF1O/ASCL1/SDO1/RP40/T4CK/RB8
SCK1/RP39/INT0/RB7
PGEC2/ASCL2/RP38/RB6
C6
10µF
X5R
10V
VCC
LED
RESET
CS
SDO
D/C
SDI
SCK
T-CLK
T_DIN
T_DO
T_CS
T_IRQ
GND
LCD panel
C8
100nF
C9
100nF
S1
S2

Fig.2. Updated *Audio Spectrum Analyser* schematic – note the addition of the MCP1700T LDO power supply section at top.

Adding these features to our design will provide a significant amount of new software that you can download, play with and expand on. The key feature being added will be the display of text; we will make use of existing open-source software libraries to provide the fonts and driver code required to implement them.

Even with just two buttons there are several ways to implement a user interface in our software. One is to do a menu system similar to the technique used by PC monitors where one button cycles through a menu list and another toggles between options. Another method would be to provide specific functions to each button. We have chosen the latter route because it will be simpler to implement. The specification for the implementation can be described as follows:

Button A

- On each press of the button, cycle between peak frequency display, peak trace hold, none
- When active, the peak frequency will be displayed in the format *xxxx* Hz directly above the peak signal.

Button B

- Pressing once will display the frequency span for three seconds
- Pressing the button again while the span is being displayed will cycle through the different span settings.

The display will be updating quite frequently, so it's possible that the display of the peak frequency value may be a little difficult to read – it will be interesting to see it in action! If it does pose a problem we can look at reducing the display refresh rate down to perhaps once per second.

Increasing the maximum frequency span of the device to 20kHz will require some component value changes on the input filter from the microphone, which we will cover in the next article. To be clear, this is not going to be a high-precision measurement device; the microphone will have poor performance above 16kHz or so, and we will not be calibrating the circuit – so consider this as a 'for indication only' design.

Let's go back and look at the circuit changes we will implement next month.

Power supply

Our main change to the electronics will be the addition of a low-noise LDO regulator. The original design used an efficient switching regulator that could operate from batteries with a voltage covering 1.5V to 5.5V. While that is great for battery life, switching regulators generate lots of noise, and when studying the circuit in operation we did see considerable noise on the microphone input signal and the output from the operational amplifier in the microcontroller. While that did not appear to impact performance, we would like to improve the design. So we have decided to implement two regulators on the PCB board – leaving you the choice of which to use. We will add a simple three-pin LDO regulator, which is not very efficient, but will generate significantly less noise. You can fit either regulator circuit, as shown in the schematic.

Adding an LDO in this project is not really a bad choice. True, LDOs are very inefficient, and dump wasted energy as heat – which can be bad for the component and other devices surrounding it. However, in practice, it depends on just how much current you are drawing – in low-power product designs, an LDO can be more efficient that a switching regulator. This is why many low-power microcontrollers incorporate both regulator technologies and switch between them depending on the system current consumption at the time. In our design we draw 120mA at 3.3V, which is a little too high to use that argument, but it's low enough that in use we will not notice the poor efficiency.

For the new LDO design we opted for a small three-pin SOT-23 regulator from microchip because it is cheap and readily available (plus we had some in the lab.) We chose to deviate slightly from its datasheet recommendations for circuit design by choosing 10µF bypass capacitors rather than 1µF. Why? Simply because we used 10µF in the previous design, and for regulation a slightly larger value would be better. Feel free to use 1µF if that is more convenient.

The final design changes, including the design corrections from last year, are shown in Fig.2. Those final design corrections were:

- Pin 2 of the CPU now connects to R4, not Pin 5
- Pin 3 of the CPU connects to pin 17
- The signal that was on pin 17 now connects to pin 18.

Before committing this design to a PCB it's worth running some tests, so as we had the LDO regulator IC in the lab, we removed the original regulator circuit (the inductor and the switching regulator) and 'birds nest' soldered the alternative regulator in place. It's a bit fiddly, but with the aid of our PCB microscope we covered in last month's *Practically Speaking* column, we were able to come up with a working solution shown in Fig. 3. By soldering two pins of the regulator over one of the original input bypass capacitors it took just a single wire to complete the circuit.

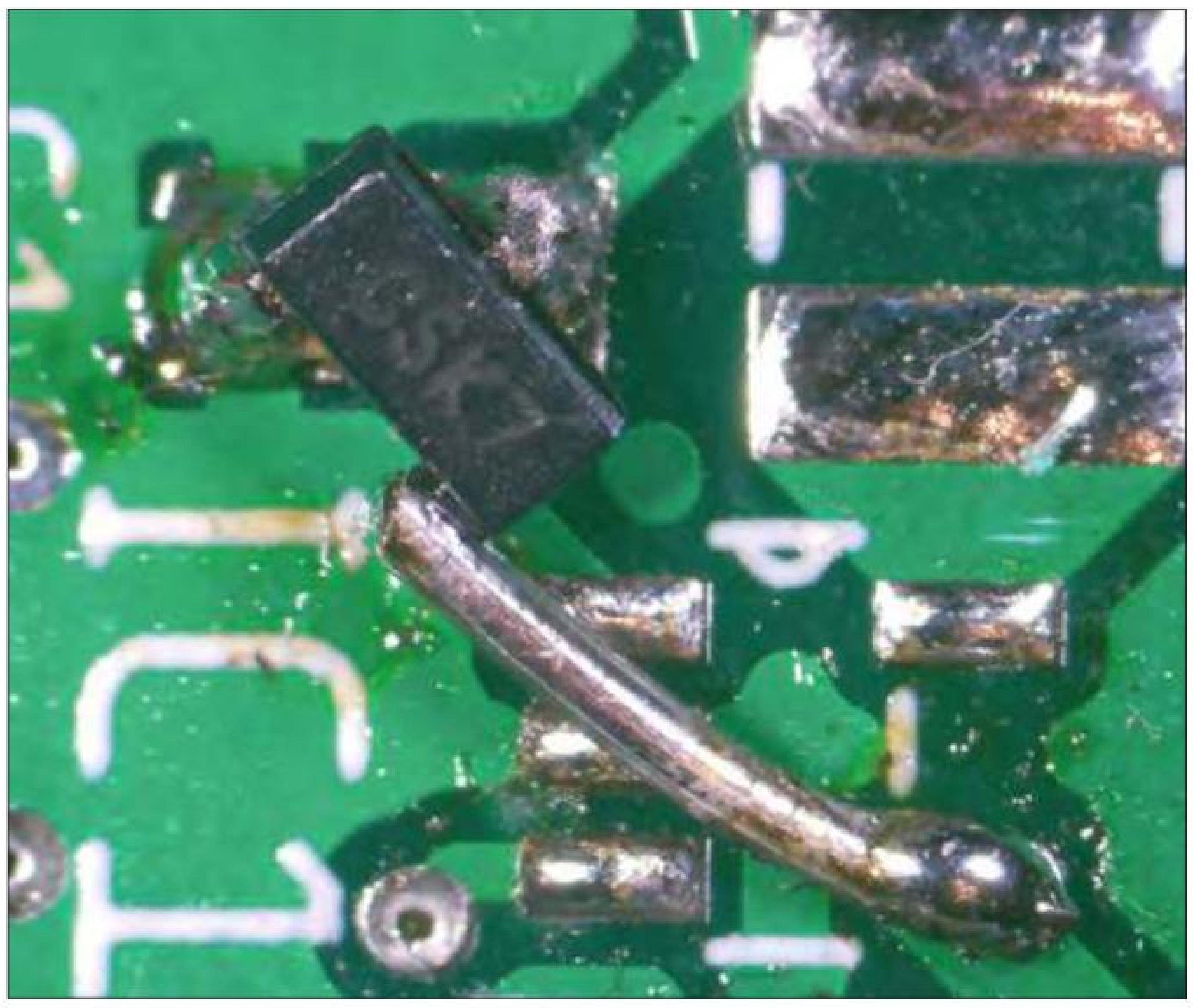


Fig.3. Testing the new LDO power supply option on the PCB.

Unsurprisingly, the circuit draws 120mA with a 6V input supply and continues to draw 120mA down to 3.3V. Where we must be careful in this design implementation is with the circuit being powered by a 'high' battery voltage. At 6V input (four new AAA alkaline batteries), the regulator is dumping 6.0 – 3.3V, or 2.7V, out of the regulator as heat. With a constant 120mA current draw, this equates to 324mW. That energy is being dissipated by a very small package, so we can expect it to get warm. If it gets too warm, that will be a problem – the device could fail.

To test this we hooked the circuit up to a bench supply and ran it at 6V while monitoring the chip temperature with a low-cost thermal sensor. These devices cost less than ten pounds and are readily available (eg, eBay item 383337313555.) These sensors are not terribly accurate but are perfectly suited for this kind of application, where we are looking for a general indication of heat rise. After an hour of operation the regulator settled at 35°C, just 10°C above room ambient. That's perfectly acceptable, but we will still take care during PCB design to provide as large an area of copper as possible on the ground pin to help dissipate the heat being generated.

That completes the circuit changes in preparation for the next article. To complete the physical design we will add a simple on-off switch to the battery, but we will mount that in the enclosure – the design of which will be covered in another article.

Speaking of enclosures brings us to the final point – what batteries to use? We need to determine which of the two power supply options to fit, as this will impact enclosure design.

To recap, with this new design we can support two battery voltage ranges:

- 1.5V – 5.5V with the noisy but efficient switching regulator; works with two or three alkaline batteries
- 3.3V – 6V with the quiet but inefficient LDO regulator. This will work with four alkaline batteries.

If you are trying to build something very small, the former option works well with, say, two AAA batteries. The second option will require four AA or AAA batteries. Using four AAA batteries you can expect around six hours use; and with two AAA batteries a little longer. We feel that with the low cost of alkaline batteries and the benefits of a low-noise regulator, choosing the LDO regulator and four AAA batteries will be the best choice for the enclosure design. You can of course choose your own configuration. If we can, we will make space for both battery holder options in our enclosure design.

Enclosure considerations

When we come to the 3D-printed enclosure design we will be looking at providing the following features:

- Include legends for the buttons, indicating their function
- Provide a recess for the on/off switch, so we do not need to drill or cut the plastic
- Provide a hole for the LCD, so no need for accurate drilling, cutting or 'dremel-ing'
- Angle the enclosure so it can be easily visible on the desktop
- Provide easy access for replacement of the batteries.

Next time

In the next article we will have the PCB design completed and the board back from PCBWay, do we will be able to discuss the software changes and review the improvements as a result of our PCB changes. If time permits, we will present the 3D-printed enclosure design.

GET THE LATEST COPY OF OUR TEACH-IN SERIES AVAILABLE NOW!

Order direct from Electron Publishing

PRICE £8.99

(includes P&P to UK if ordered direct from us)

Teach-In 9 – Get Testing!

This series of articles provides a broad-based introduction to choosing and using a wide range of test gear, how to get the best out of each item and the pitfalls to avoid. It provides hints and tips on using, and – just as importantly – interpreting the results that you get. The series deals with familiar test gear as well as equipment designed for more specialised applications.

The articles have been designed to have the broadest possible appeal and are applicable to all branches of electronics. The series crosses the boundaries of analogue and digital electronics with applications that span the full range of electronics – from a single-stage transistor amplifier to the most sophisticated microcontroller system. There really is something for everyone!

Each part includes a simple but useful practical test gear project that will build into a handy gadget that will either extend the features, ranges and usability of an existing item of test equipment or that will serve as a stand-alone instrument. We've kept the cost of these projects as low as possible, and most of them can be built for less than £10 (including components, enclosure and circuit board).

FREE COVER-MOUNTED CD-ROM

On the free cover-mounted CD-ROM you will find the software for the *PIC n' Mix* series of articles. Plus the full *Teach-In 2* book – *Using PIC Microcontrollers – A practical introduction* – in PDF format. Also included are Microchip's *MPLAB ICD 4 In-Circuit Debugger User's Guide*; *MPLAB PICkit 4 In-Circuit Debugger Quick Start Guide*; and *MPLAB PICkit4 Debugger User's Guide*.

ORDER YOUR COPY TODAY JUST CALL
01202 880299 OR VISIT www.epemag.com

PE Practical Electronics

BACK ISSUES

EPE EVERYDAY PRACTICAL ELECTRONICS

DECEMBER 2019

PROJECTS • Extremely Sensitive Magnetometer • Useless Box! • Four-channel High-current DC Fan and Pump Controller • Colour Maximite Computer – Part 2

FEATURES • The Fox Report • Techno Talk • Net Work • Circuit Surgery • Using Stepper Motors • Audio Out • Make it with Micromite • Max's Cool Beans • Electronic Building Blocks

JANUARY 2020

PROJECTS • Audio DSP – Part 1 • Isolated Serial Link • Four-channel High-current DC Fan and Pump Controller – Part 2 • Colour Maximite Computer – Part 3

FEATURES • The Fox Report • Techno Talk • Net Work • PIC n' Mix • Using Stepper Motors • Circuit Surgery • Audio Out • Make it with Micromite • Max's Cool Beans • Electronic Building Blocks

FEBRUARY 2020

PROJECTS • Audio DSP – Part 2 • Motion-Triggered 12V Switch • USB Keyboard and Mouse Adaptor for Micros • Using Cheap Asian Electronic Modules – Part 21 • Colour Maximite Computer – Part 4

FEATURES • The Fox Report • Techno Talk • Net Work • Practically Speaking • Using Stepper Motors • Circuit Surgery • Audio Out • Make it with Micromite • Max's Cool Beans • Electronic Building Blocks

BACK ISSUES – ONLY £5.80

£5.80 per issue for UK incl p&p ■ £6.75 Europe Air Mail ■ £7.50 ROW Air Mail

We can supply back issues of PE/EPE by post.

We stock magazines back to 2006, except for the following:

2006 Jan, Feb, Mar, Apr, May, Jul
2007 Jun, Jul, Aug
2008 Aug, Nov, Dec
2009 Jan, Mar, Apr
2010 May, Jun, Jul, Aug, Oct, Nov
2011 Jan
2014 Jan
2018 Jan, Nov, Dec
2019 Jan, Feb, Apr, May, Jun

Issues from Jan 1999 are available on CD-ROM / DVD-ROM

If we do not have a a paper version of a particular issue, then a PDF can be supplied for the same price. Your email address must be included on your order.

Please make sure all components are still available before commencing any project from a back-dated issue.

ORDER FORM – BACK ISSUES

☐ Back issues required (month / year) . . . .

Name . . . .

Address. . . . .

. . . . .Tel: . . . .

Email . . . .

☐ I enclose cheque/PO to the value of £ . . . .

☐ Please charge my Visa/Mastercard £ . . . .

Card No . . . .

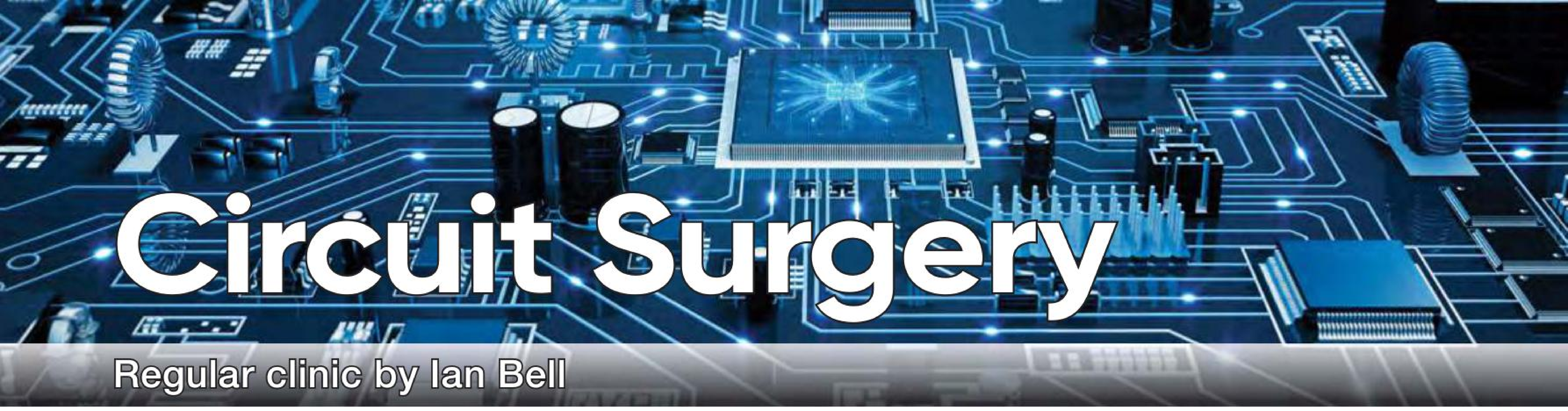
Valid From . . . . Card Expiry Date . . . .

Card Security Code . . . . (Last three digits on or under the signature strip)

SEND TO: **Practical Electronics, Electron Publishing Limited**
113 Lynwood Drive, Merley, Wimborne, Dorset BH21 1UU

Tel: 01202 880299 **Email**: stewart.kearn@wimborne.co.uk **On-line Shop**: www.epemag.com

Payments must be in £ sterling – cheque must be drawn on a UK bank and made payable to 'Practical Electronics'. All items normally posted within seven days of receipt of order. Copy this form if you do not wish to cut your issue.



Circuit Surgery

Regular clinic by Ian Bell

Strain gauge circuit revisited

In the November 2019 issue we looked at strain gauges and differential signals in an article inspired by a post on the EEWeb forum from Scott Siler, who wrote: 'I'm an aerospace engineer working on a side project privately to gather data from a strain gauge using an NI DAQ (a data acquisition device from National Instruments). The output is 0-36mV DC so I need a gain of 3 to better utilise the DAQ. My EE experience is limited to a couple of classes I took during my bachelor's degree. I have an AD8628 op amp and built a basic, non-inverting negative feedback circuit, as shown in the attached picture (see Fig.1). The rail voltage is 0-3V DC. It does not seem to have any gain, but instead the output voltage is actually lower than the input.'

We received an email from John Ellis, who though that I had not diagnosed the issues with the Scott's circuit in sufficient detail. John wrote: 'I read Ian Bell's article on strain gauges and differential amplifiers with interest. I thought, however, that the article might have provided a little more information about the problem faced by Scott Siler, which may have been more help to your readers in designing their own sensor circuits.'

John also provided his own analysis of the circuit, deriving the asymmetrical gain formula for the circuit in Fig.1 (which shows that it is not a good differential amplifier) and also pointing out possible problems with the common-mode input voltage (which may be the main reason it did not work). We will look at these aspects of op amp circuit design in this article.

In response to John's criticism, there are a few reasons why we do not necessarily look at the specific details of problems posted on the forum, but cover a topic in more general terms, as we did with the November 2019 article. First, the full details are not always posted on the discussion thread – this was the case here to some extent – we do not have a complete schematic of Scott's circuit (although we can guess) and we do not know all the details of the load cell used. Second, specific problems are often solved by forum contributors long before an article can be published, so it makes sense for *Circuit Surgery* to take a broader look at the topic. Third, there are often several themes that can be developed from the issues discussed in the forum thread, and often there will not be space to address them all in a single article. Thus, we often have multi-part articles and sometimes return to a topic on request – which is exactly what we are doing here.

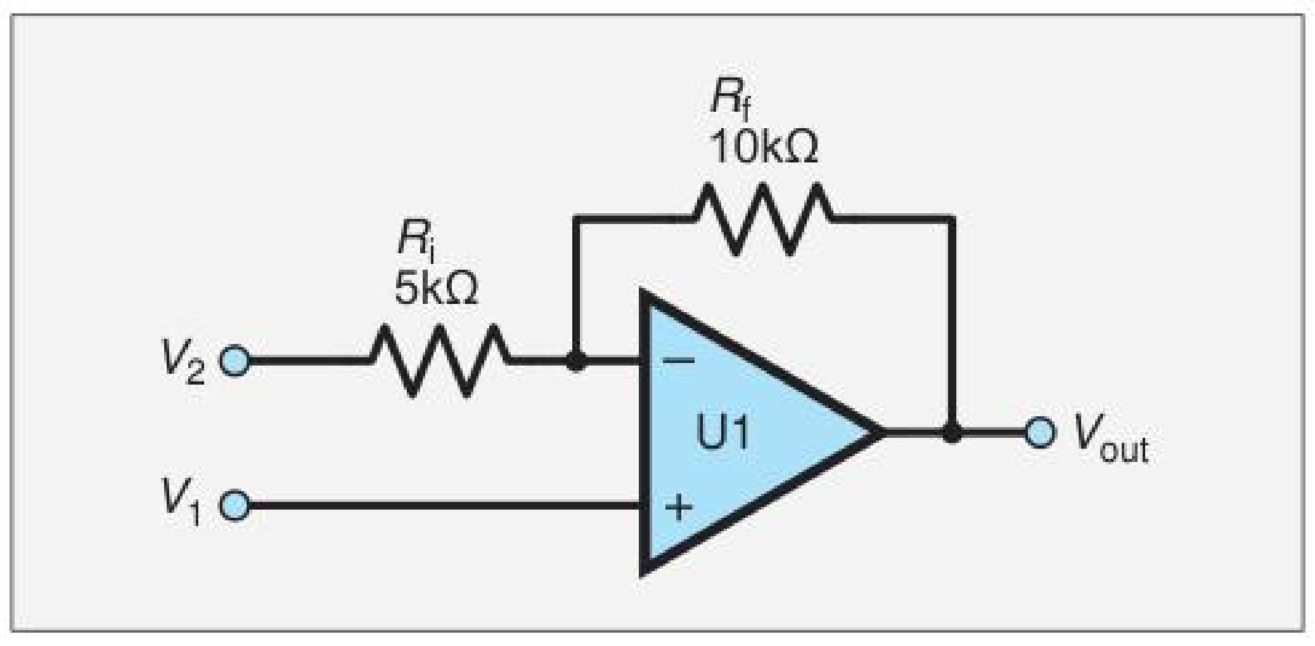


Fig.1. Scott's circuit, as discussed in this article and *Circuit Surgery*, November 2019.

For the benefit of readers who do not have the November 2019 article to hand, we will briefly recap strain gauges, load cells and differential and common-mode signals to provide background and context for discussion of the op amp circuits.

Strain Gauges

Strain is a measure of the deformation (change in size or shape) of an object. A strain gauge is a sensor whose electrical resistance changes with deformation. It typically comprises a thin, flexible, insulating foil that supports a long conductive strip, typically in a zig-zag pattern (see Fig.2) with contacts to connect the device to a measurement circuit. Strain gauges are often built into larger devices, called 'load cells', in which strain gauges are attached to a specifically designed metal body that deforms when force is applied to the device. Load cells have many industrial applications in force and weight measurement.

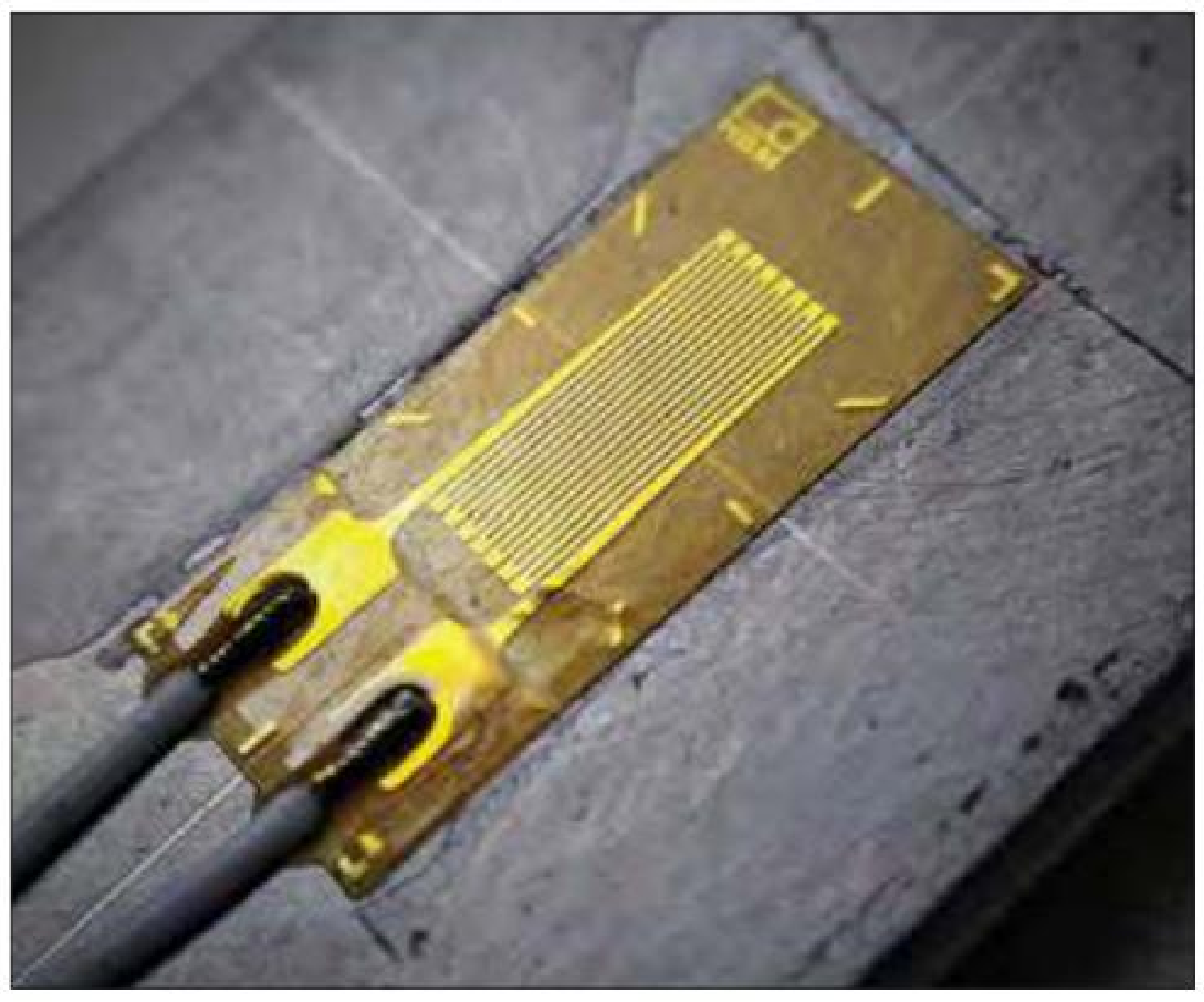


Fig.2. A typical strain gauge cemented to the substrate under measurement.

Strain gauges are typically used in Wheatstone bridge circuits, as shown in Fig.3. The bridge comprises two potential dividers in parallel, with the output voltage being the difference between the two divider voltages. The differential voltage from a bridge does not have the offset associated with a simple potential divider, so it can be amplified without the offset causing the amplifier to saturate. The circuit in Fig.3 has a single strain gauge with three fixed resistors, a variety of possible physical measurement scenarios use multiple stain gauges. The bridge can contain one, two, three or four strain gauges depending on the setup used. In some situations additional resistors may be used (eg, between the excitation voltage and bridge) to fine tune the behaviour of the circuit.

Differential and common-mode signals

A differential signal is carried on two wires other than ground, so we can observe the voltage on each wire individually (eg, V_1 and V_2). The actual signal is equal to the difference in the

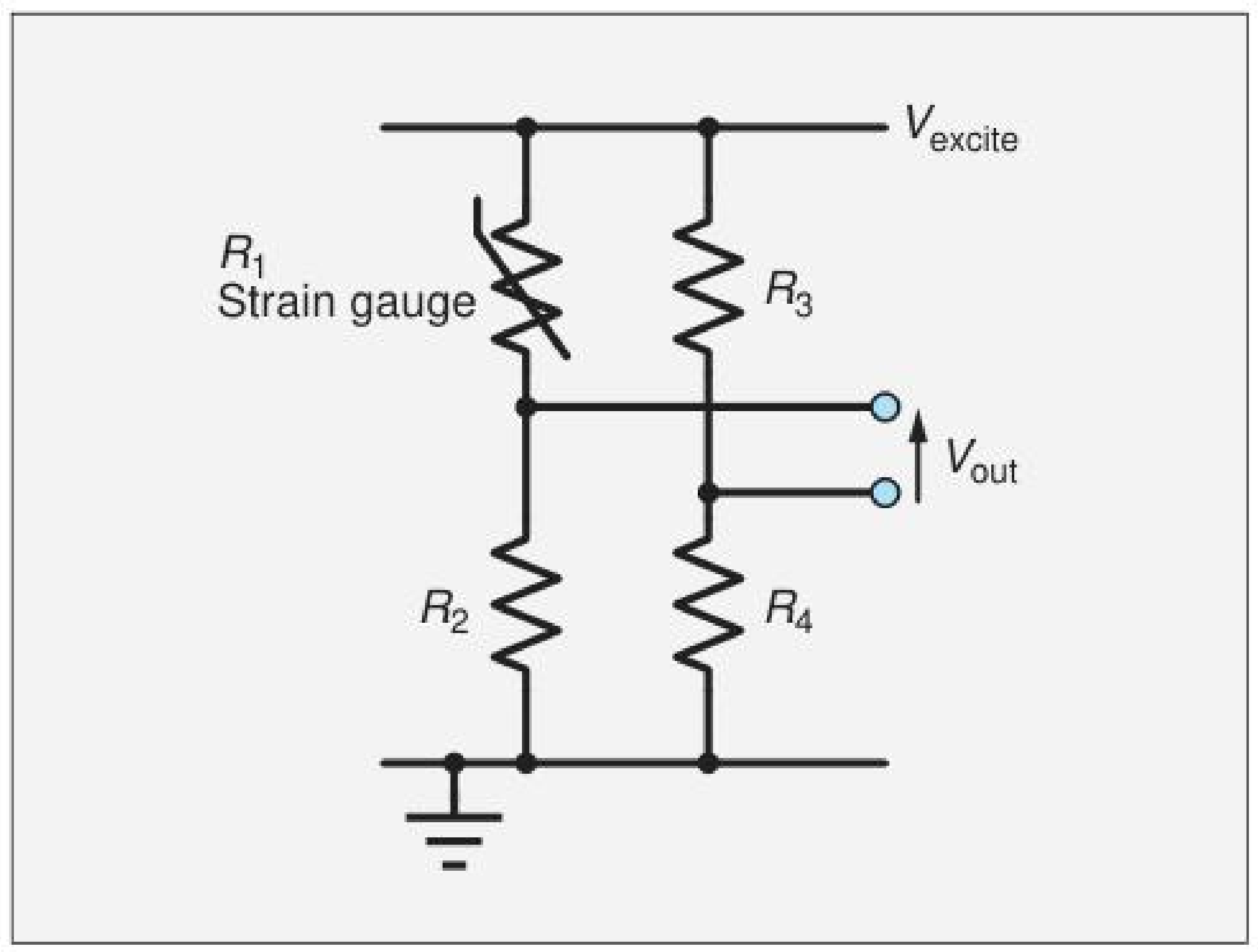


Fig.3. A common strain gauge bridge circuit – in this example R1 is the strain gauge, but other arrangements are used.

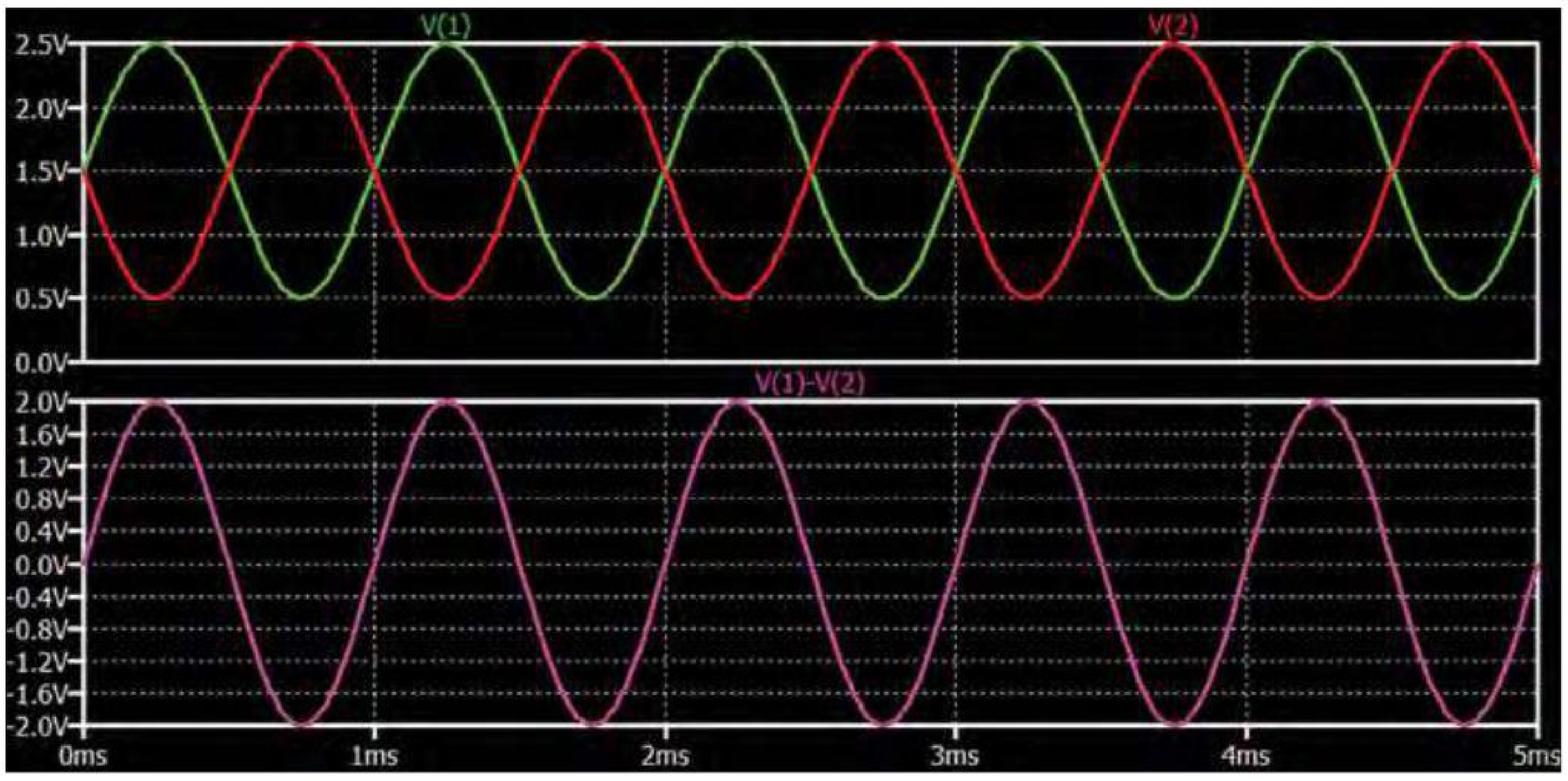


Fig.4. Example differential signal: 2V peak, 1kHz sinewave with 1.5V DC common mode. The upper traces (red, green) are the individual voltages. The lower trace (magenta) is the differential signal.

voltages between the two wires, each measured with respect to ground.

So if the two voltages on the wires are V_1 and V_2, then the differential signal is ($V_1 - V_2$). However, in order to fully describe a differential signal we need to state two things – the differential signal itself, and the voltage they have in common – called the 'common-mode voltage'. The common-mode voltage is the average of the voltage on the two wires ($V_1 + V_2$)/2. An 'ordinary' signal carried on a single wire is described simply by its voltage with respect to ground. Such signals may be referred to as 'single ended'; where it is necessary to make a distinction from differential signals in a discussion.

Fig.4 shows an example differential signal – this is a 1kHz sinewave with a peak voltage of 2V and common-mode component which is 1.5V DC. The upper plot shows the individual signals (V_1 and V_2) and allows the 1.5V common-mode signal to be observed. The lower plot shows the differential signal as a single waveform.

Differential amplifiers

A differential voltage amplifier is a circuit which has two inputs and amplifies the voltage difference between them – thus, it amplifies the differential signal at its input. If its inputs are V_1 and V_2 and its differential voltage gain is A_d, then its output is $A_d(V_1 - V_2)$. Its output may be differential (in which case it is referred to as a 'fully differential' amplifier), or single ended. A standard op amp is a differential amplifier with a single-ended output. However, op amps are very commonly used to build amplifiers circuits which have both single-ended inputs and outputs – these are the well known inverting and non-inverting configurations (see Fig.5 and later discussion).

An ideal differential amplifier only amplifies the difference between V_1 (non-inverting input) and V_2 (inverting input). One way of looking at what the differential amplifier does is to think of each input as having a separate gain. The output is then made up from the non-inverting input signal times its gain (A_1) minus the inverting input signal times its gain (A_2). We can write this as an equation as follows:

$$V_{out} = A_1V_1 - A_2V_2$$

If $A_1 = A_2 = A_d$ then this equation becomes the ideal case of $V_{out} = A_d(V_1 - V_2)$, which we noted above. Ideally, the gains for the two inputs are equal; however, this is not the case for real differential amplifiers. With a little algebraic manipulation we can rearrange the above equation so that it includes the term ($V_1 - V_2$) plus some other terms. It is useful to do this because we can then see an 'ideal part' – the differential gain multiplying ($V_1 - V_2$), and an 'error part' – the rest of the equation. We get

$$V_{out} = \frac{(A_1 + A_2)}{2}(V_1 - V_2) + (A_1 - A_2)\frac{(V_1 + V_2)}{2}$$

in which we can see that ($A_1 + A_2$)/2 corresponds with A_d in the ideal equation, and is the average of the two gains. The other term contains ($V_1 + V_2$)/2, which is the common-mode input signal (the average of the input voltages), which is multiplied by a gain which we refer to as the common-mode gain, A_{cm}. The common-mode gain is equal to the difference between the two individual gains, so in the ideal case, where the two gains are equal, the value of ($A_2 - A_1$) is zero and the common-mode input has no effect on the output.

We can rewrite the above equation as:

$$V_{out} = A_d(V_1 - V_2) + A_{cm}\frac{(V_1 + V_2)}{2}$$

Where A_d is the differential voltage gain and is A_{cm}, the common-mode gain.

The smaller the influence of common-mode signals on the differential amplifier, the better the amplifier. The ability of an op amp to reject common-mode signals is expressed as the ratio of the differential and common-mode gains; this is called the 'common-mode rejection ration' (CMRR), which is often expressed in decibels as follows:

$$CMRR = 20\log_{10}\left|\frac{A_d}{A_{cm}}\right| \text{ dB}$$

The CMMR of the AD8628 used in Scott's circuit is very high at 130dB (the common-mode gain is more than three-million-times smaller than the differential gain).

Op amp amplifiers

It is useful to consider Scott's circuit in terms of the general differential amplifier properties discussed above. However, first we will briefly describe the most basic and commonly used op amp amplifier circuits – the inverting or non-inverting amplifier configurations, as shown in Fig.5. In both cases negative feedback is applied via a pair of resistors, which act as a potential divider feeding a fraction of the output voltage back to the inverting input. The gain of these circuits is related to the ratio of the resistor values, which sets the proportion of the output fed back by the potential divider. The amplifier as a whole is either inverting (negative gain) or non-inverting (positive gain) depending on whether the input signal is routed to the inverting or non-inverting input.

Scott's circuit (Fig.1) is similar to the circuits in Fig.5 – the feedback arrangement is the same, but it has two inputs instead

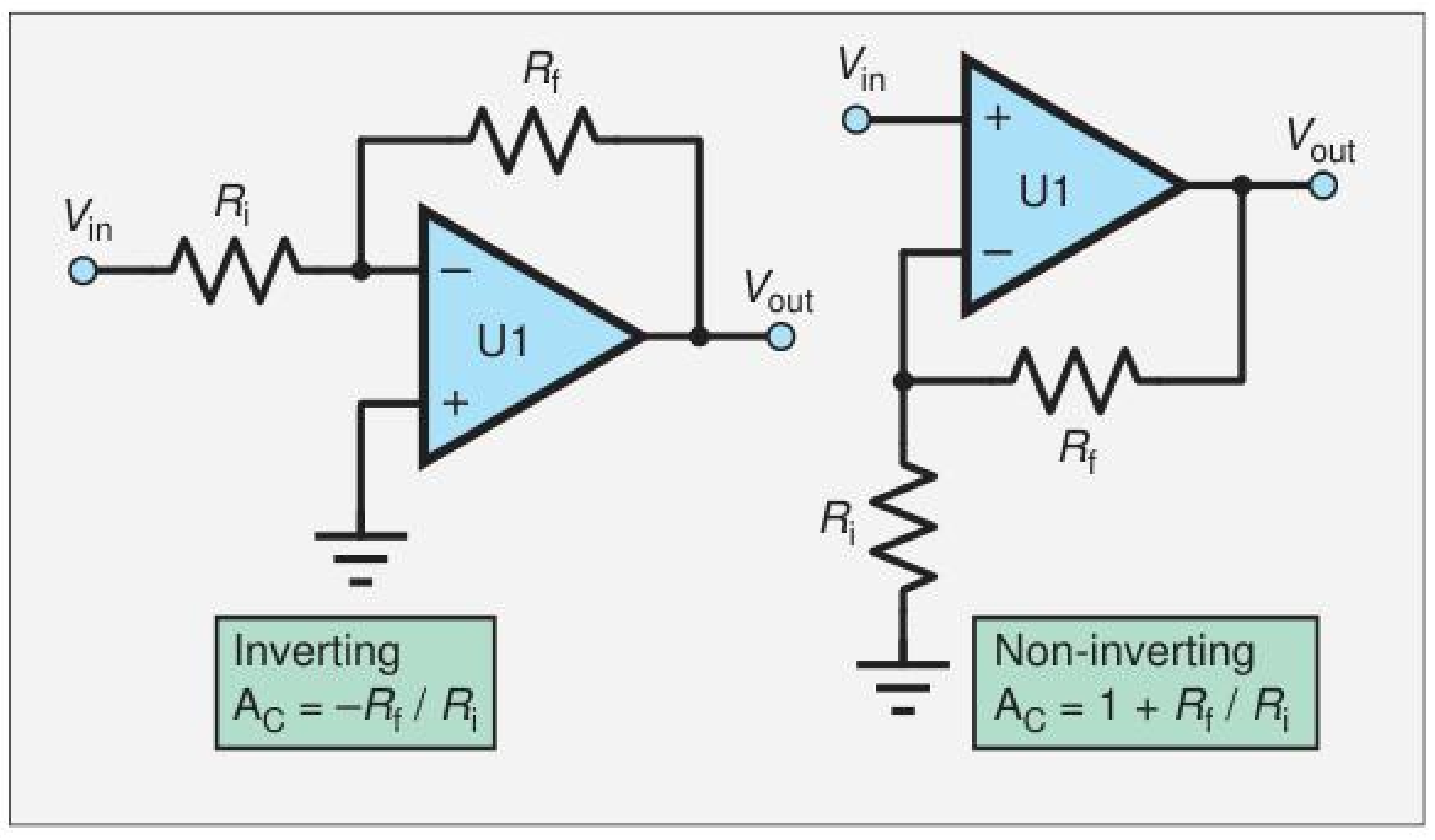


Fig.5. Basic op amp amplifiers: inverting (left) and non-inverting (right).

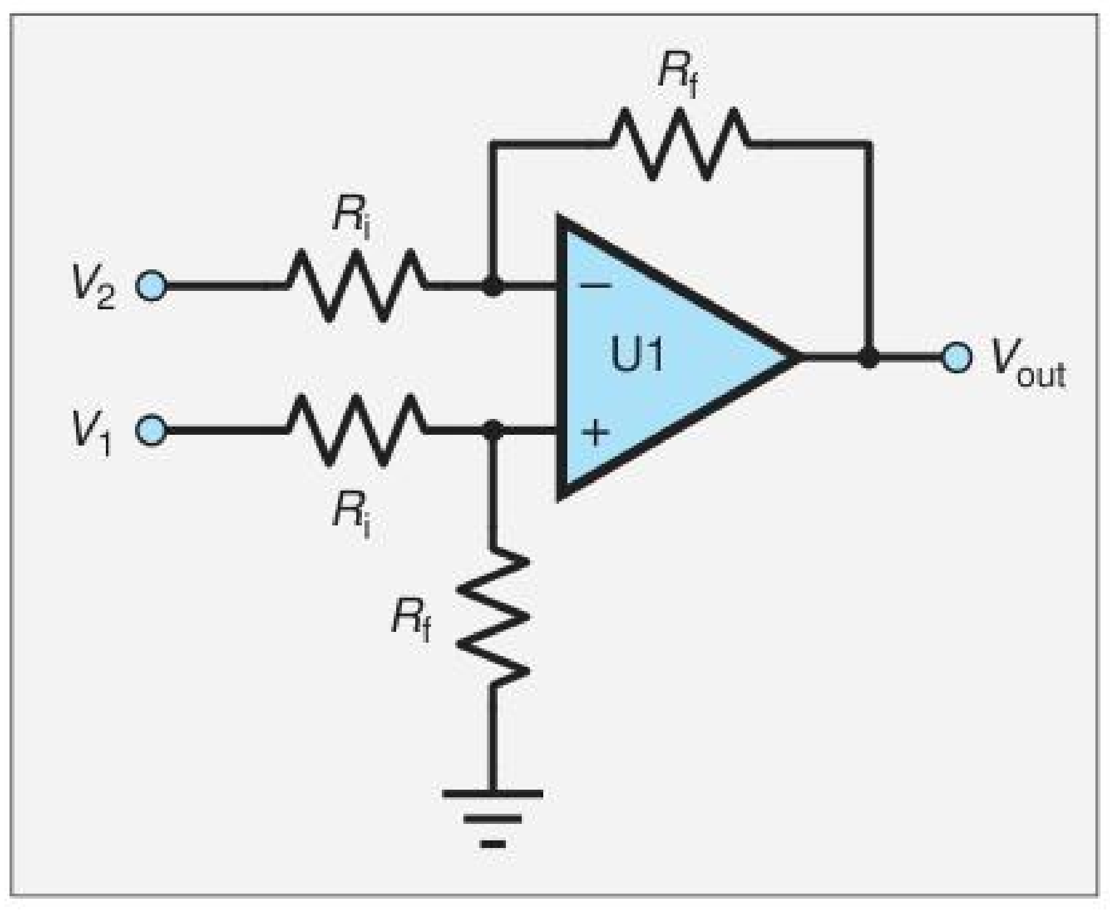


Fig.6. The standard differential op amp amplifier.

of one. In fact, Scott's circuit is like a combination of the two basic amplifier configurations. If we ground the V_1 input to Scott's circuit we get an inverting amplifier with V_2 as its input. If we ground the V_2 input we get a non-inverting amplifier with V_1 as its input.

Superposition

Having identified this relationship between Scott's circuit and the basic op amp configurations, we can use some circuit theory called the 'superposition theorem' to obtain a formula for its gain. The superposition theorem states that for a linear circuit we can find a voltage or current of interest by taking all the sources in turn, setting all the other sources to zero, working out the relevant circuit values with just that one source active, repeating this for all the sources, and then adding the results from the individual sources to get the required value.

For Scott's circuit we have two sources – the inputs V_1 and V_2. We don't have to worry about the supplies – they are fixed DC voltages that do not carry signals, and (at least ideally) do not change the output voltage or gain, as long as the op amp is operating normally. The output due to V_2 with V_1 grounded is:

$$\left(-\frac{R_f}{R_i}\right)V_2$$

The output due to V_1 with V_2 grounded is:

$$\left(1+\frac{R_f}{R_i}\right)V_1$$

Adding these together we get:

$$V_{out} = \left(1+\frac{R_f}{R_i}\right)V_1 - \left(\frac{R_f}{R_i}\right)V_2$$

We can compare this equation to the non-ideal differential amplifier with separate gains (A_1 and A_2), as described above; and finding the differential gain as the average of A_1 and A_2, we get:

$$A_d = \left(\frac{R_i + 2R_f}{2R_i}\right)$$

A bad circuit?

With Scott's resistor values the differential gain is: (5 + (2 × 10)) / (2 × 5) = 2.5.

Similarly, we can find the common-mode gain from $A_1 - A_2$ which is simply unity. Thus the CMMR is just 2.5 (about 8dB) and is more than a million times worse than that of the op amp itself! Using a high performance op amp does not guarantee a high performance circuit.

Previous discussion has dismissed Scott's circuit as not being a differential amplifier, but this is not strictly true – it is a differential amplifier, just a very poor one. The differential gain is not the value required by Scott (3), but this could be fixed with different resistors. This incorrect differential gain value is unlikely to be the reason that the circuit did not perform as Scott hoped.

The circuit configuration in Fig.1 may be a bad differential amplifier, but it is not a useless circuit in other contexts. A common use for this circuit is to simultaneously amplify a signal and shift its DC level. For example, consider a signal with range ±0.5V, which needs to be input to an ADC with a 0-5V input. A gain of 5 is required to map the ±0.5V range to 0-5V, but it is also necessary to shift the DC level from 0V to 2.5V. The circuit configuration in Fig.1 powered from a ±5V or greater supply can handle the required signals. If we connect the signal to the non-inverting input (V_1 in Fig.1) with a resistor ratio satisfying $1 + R_f/R_i = 5$, that is $R_f/R_i = 4$, will get the required gain of 5 with respect to this input. The gain with respect to the inverting input will then be –4 ($-R_f/R_i$). A DC input at the V_2 input in Fig.1 will be amplified by –4, so if we connect this input to a DC level of 2.5/–4 = –0.625V (for example, obtained via a potential across the negative supply rails) this will produce the required +2.5V DC output when the signal is 0V.

Alternative circuits

As we discussed in the previous article there is a standard op amp circuit for a 'proper' differential amplifier (see Fig.6). Comparing this with Fig.1 we see that it is essentially the same circuit with a potential divider at the V_1 input. The divider attenuates the V_1 input to compensate for the higher gain via the inverting input in Fig.1. The differential gain of the circuit in Fig.6 is simply R_f/R_i – note that the two pairs of resistors have the same values. As discussed in the previous article, this circuit only achieves high CMRR (limited by the op amp) if the pairs of R_f and R_i resistors both have very accurately matched values – so it is difficult to achieve very high CMRR with discrete resistors. Another issue with this circuit is its relatively low input impedance, which means that it may load signal sources such a sensors, potentially leading to measurement errors. As discussed in the previous article a solution to this is to use an instrumentation amplifier circuit or IC.

Common-mode input

As indicated above, the poor performance of Scott's circuit as a differential amplifier is probably not the reason that it failed to work. The gain of 2.5 would still provide a reasonable output (not 'lower than the input'). The reason is more likely to be related to the common-mode input applied to the circuit in Scott's system. Assuming the load cell Scott used is similar to Fig.3 and that all four bridge resistors have approximately the same value (as is typical) then the signal from the load cell will be a small differential voltage (Scott states 0 to 36mV) with a common-mode voltage of half the load cell excitation voltage (due to the almost-equal resistances in the potential dividers). Scott stated that his load cell is operated on 12V, but did not provide further details.

If we assume the simplest case – the load cell is connected to a 12V supply relative to ground – then its common-mode output would be 6V. If we just consider the gains calculated earlier we might expect the output of Scott's circuit to be 2.5 times the differential signal (about 0 to 90mV), with a common-mode output of one times the 6V input – that is 6V. There is a problem here – Scott states that the AD8628 op amp circuit (Fig.1) is operating from a 3V supply (and its maximum supply is 5V). If our assumptions about the load cell power are correct then the common-mode voltage into the circuit in Fig.1 would be well beyond the supply voltage. Typically, op amps do not work under such conditions.

Op amps have a characteristic called 'common-mode input range' which specifies the extremes of common-mode input at which they will work correctly. Some op amps (including the AD8628) can cope with common-mode signals equal to the supply voltage – often referred to as a 'rail-to-rail input' capability. We need to check what the datasheet says about the AD8628's common-mode input range and its behaviour with overvoltage inputs.

The AD8628

The AD8628 is made by Analog Devices and described as a zero-drift, single-supply, rail-to-rail input/output op amp. It is not an ordinary op amp in that it achieves its high precision through a combination of auto-zeroing and chopping techniques (patented by Analog Devices). Analog Devices state that this allows the op amp to maintain its low offset voltage over a wide temperature range and over its operating lifetime, and that it has low noise compared with other auto-zero amplifiers. It is available in a variety of packages – the common 8-lead SOIC and MSOP forms, and the less usual (for op amps) 5-lead TSOT and SOT-23 packages (as used by Scott). See Fig.7 for the 5-lead pin-out. Strain gauge amplifiers are listed among the suggested applications for the device.

Auto-zeroing and chopping amplifiers are designed to deal with the difficulty of high-precision amplification of low-frequency signals (typically a few hertz

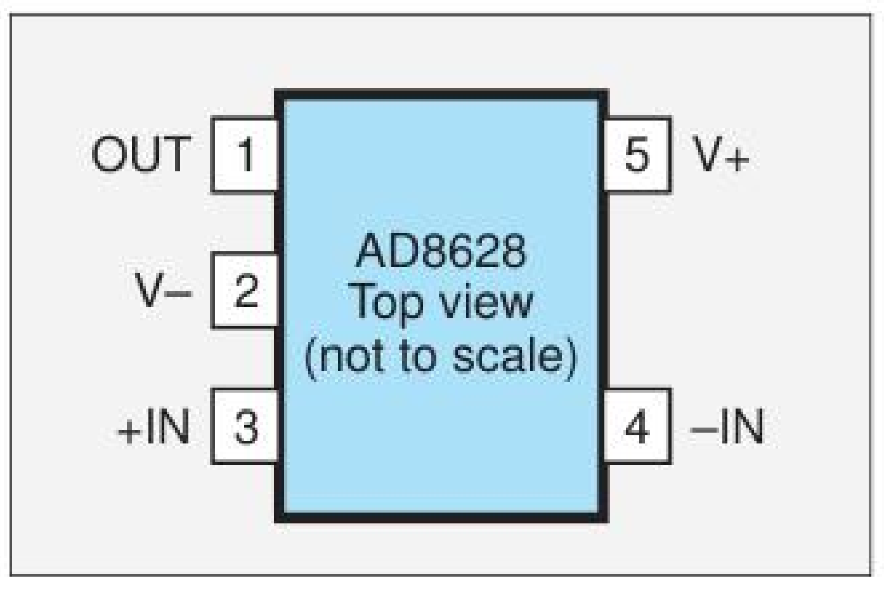


Fig.7. AD8628 pinout for 5-lead TSOT and SOT-23 packages.

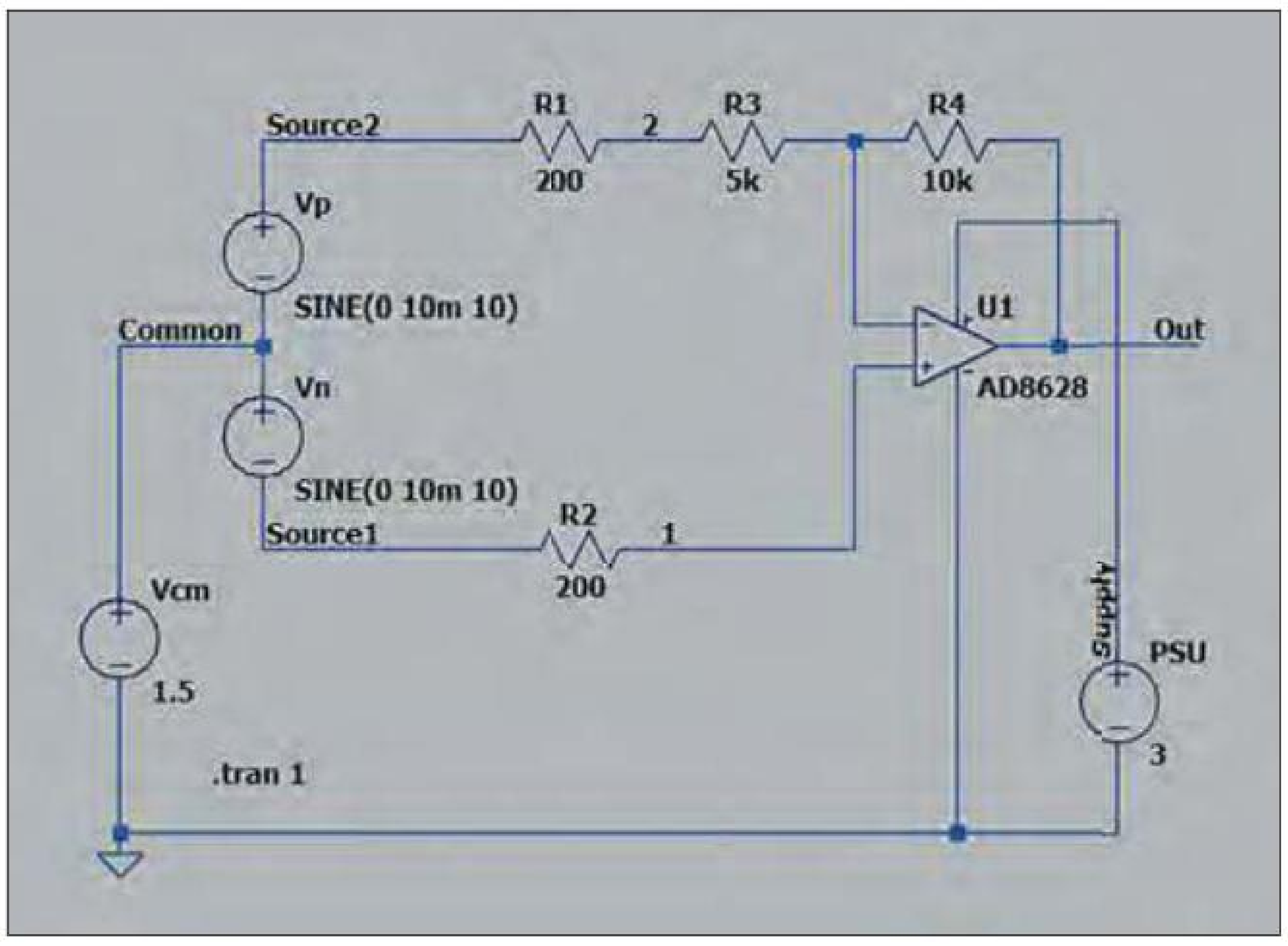


Fig.8. LTspice schematic for simulating the circuit in Fig.1.

and below). Depending on the application, these amplifiers may also be required to handle much higher frequencies, adding to the difficulty of creating suitable circuit designs. Low-frequency signals typically occur in certain sensing implications, such as strain gauges, where the measured signal changes slowly. The problem is the low-frequency noise and offsets which are inherent in electronic amplifiers and which render ordinary op amps inadequate for these tasks. The solution makes use of switching circuits and, as indicated by the dual approach taken by the AD8628, there are two basic ways to implement these circuits – auto-zero and chopping. We discussed these concepts in depth in *Circuit Surgery* in the June and July 2018 issues.

Overvoltage

The 'rail-to-rail input' part of the AD8628 description indicates that it can handle input signals close to the supply, but the datasheet goes into more detail. It states that if either input exceeds either supply rail by more than 0.3V, large currents may flow through the ESD (electrostatic discharge) protection diodes in the amplifier. These diodes are connected between the inputs and each supply rail and are normally reverse-biased. However, they can become forward-biased if the input voltage exceeds the supply voltage. If excessive current flows as a result of this the device can sustain permanent damage. In circuits where overvoltage may occur the datasheet recommends series resistors at the inputs to limit the current to 5mA. We do not know enough about Scott's system to be certain about the level of current that might have been delivered via the strain gauge – although Scott did not mention anything about his op amp being destroyed.

As well as damage when currents are not limited, common-mode overvoltage can cause strange behaviours in some differential amplifiers, specifically the output can suddenly jump in the opposite direction to the supply rail – a phenomenon known as output phase reversal. With the AD8628, limiting the overvoltage input current to 5mA should prevent this. Again, we do not have enough information to know if this occurred in Scott's system. Another potential issue with common-mode overvoltage, which applies when it occurs temporarily, is that some auto-zero amplifiers take a long time to recover when the overload is removed. The data sheet for the AD8628 states that it has a much shorter recovery time than typical auto-zero amplifiers. This problem is not relevant in Scott's case as the presumed overload is permanent.

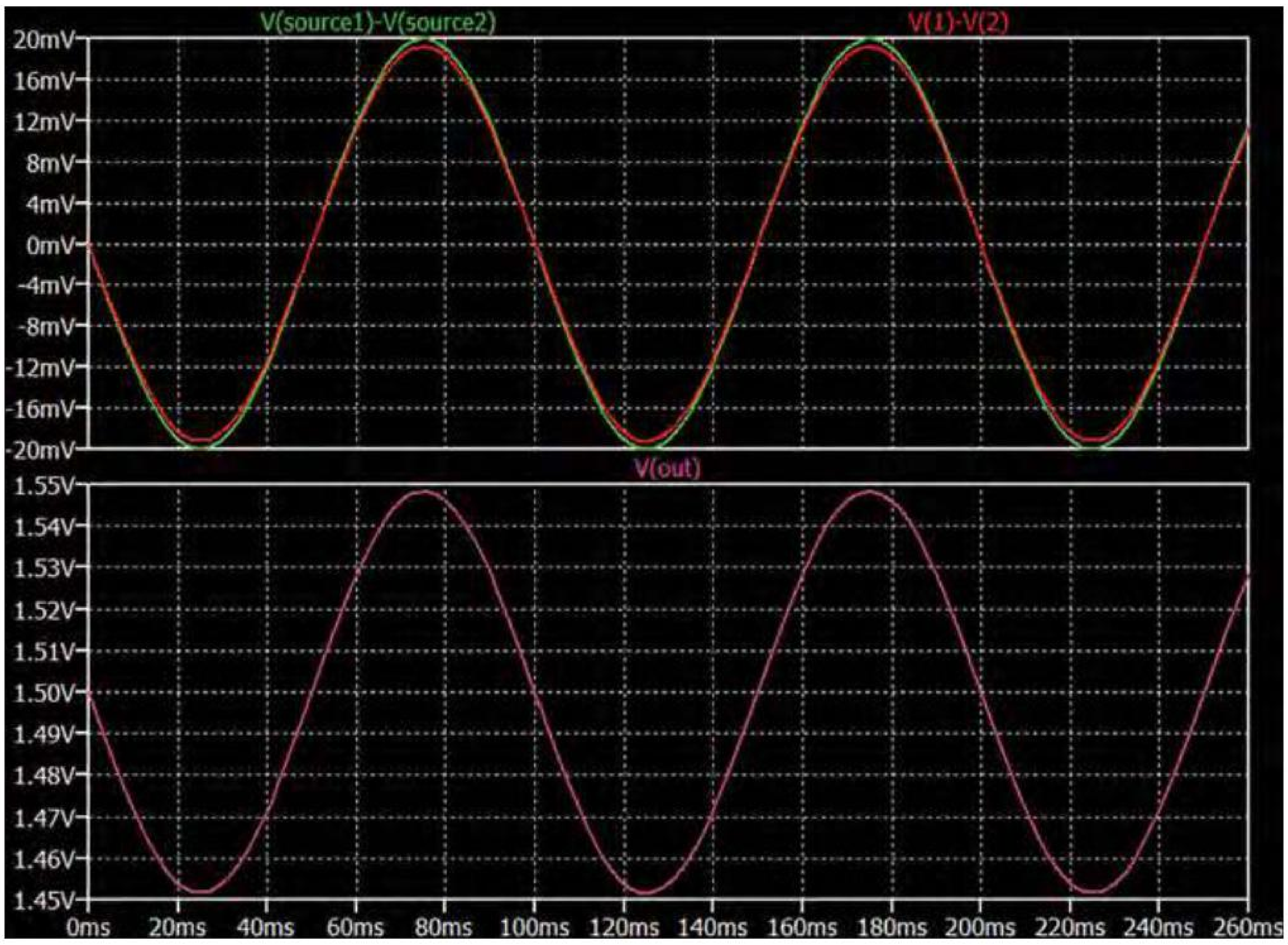


Fig.9. Results from simulation in Fig.8 of the circuit in Fig.8.

Simulation

LTspice had a model for the AD8628, so we can try a simulation. The schematic is shown in Fig.8. Here we have set up the input with specific common-mode and differential signals – 1.5V common mode, and 40mV peak-to-peak differential, are obtained with the values shown on the schematic, but these are easily changed. This approach is used as it provides direct control of the input signal and because we do not know the exact circuit of Scott's load cell. We start with the common-mode level at half the op amp supply voltage so that the circuit will operate correctly and we can confirm the gains calculated earlier. The sources have some output impedance (R_1 and R_2 at 200Ω) as the load cell would have non-zero output impedance, but values used are an arbitrary choice as we do not know the load cell details.

The results of the simulation are shown in Fig.9. We see that the input signal (`v(1) – v(2)`) is slightly lower than the source signal (`v(source1) – v(source2)`) due to loading of the source by the amplifier circuit. The differential input amplitude is about 38.3mV peak-to-peak and the output is about 96.7mV peak-to-peak, confirming the expected differential gain of 2.5 calculated above. The output is centred on 1.5V – equal to the common-mode input, confirming the common-mode gain of 1. Experimenting with the common-mode voltage indicated correct operation with 0.05V and 2.95V common-mode inputs, confirming the rail-to-rail operation. At 6V common mode the output is basically saturated at the 3V supply, with the signal showing at a level of tens of microvolts. We cannot be sure this simulation exactly replicates the situation in Scott's circuit, but if his load cell common-mode output was 6V then that is the most likely reason why his circuit did not work.

Simulation files

Most, but not every month, LTSpice is used to support descriptions and analysis in *Circuit Surgery*. The examples and files are available for download from the *PE* website.

AUDIO OUT

By Jake Rothman

PE Mini-monitor crossover for Wavecor drivers – Part 2

A pair of *PE Mini-monitor* prototypes in action – note the anti-reflection foam on the desk!

I started last month by extolling the virtues of the famous LS3/5A, describing it as, 'arguably the best mini-monitor speaker', but I qualified that with, 'it is very expensive and getting the parts can be tricky'. The purpose of the *PE Mini-monitor* project is to get most of the true LS3/5A's bang for only a fraction of the buck!

We have spent some time going through speaker parameters and crossover designs and we are nearly at the point where we can actually build a complete speaker, but first we need to complete our new crossover.

Basic circuit

For our crossover design, we arrived at the circuit shown in Fig.16 in last month's article (repeated below). As explained, it is not the final design, but it is a satisfactory foundation on which to build.

I noticed there was a bit of a response hump in the crossover region and more worryingly, a dip (at 2kHz) below 8Ω in the impedance curve (Fig.17). It is always important to keep a check on the system impedance curve when designing crossovers. The speaker designer should not make life difficult for the amplifier designer by having low dips and very rapid (reactive) changes in the impedance. This problem was fixed by adding a 56Ω damping resistor (R9) across the lower arm low-pass filter capacitor C2a. Of course, we don't want 14% of our precious bass power being wasted in this resistor, which is effectively in parallel with the bass unit. Since we only needed its effect around 2kHz, a 10µF bass-blocker capacitor (C2b) was placed in series. This capacitor does not need a low ESR, so a cheap bi-polar electrolytic can be used if you are a penny-pincher. The resulting impedance curve with the damping network is shown in Fig.18.

Phase serendipity

I got lucky here, phase is often difficult to get right in crossovers, requiring all-pass filters or moving the driver's positions around, with deep horizontal steps or undesirably wide vertical spacings. In this case, the difference in the horizontal radiation plane between the two drivers was only around 15mm, with the large hard woofer dome and recessed tweeter dome helping to

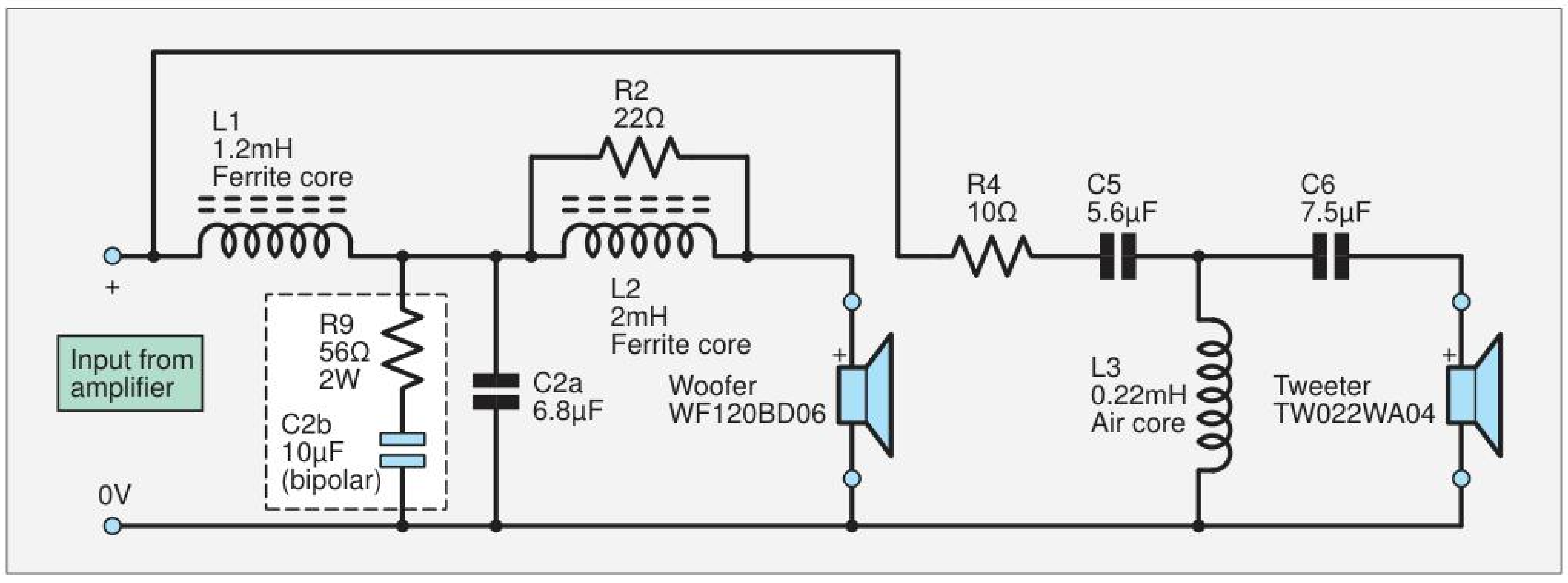


Fig.16. Complete provisional crossover from last month, plus damping network (R9, C2b).

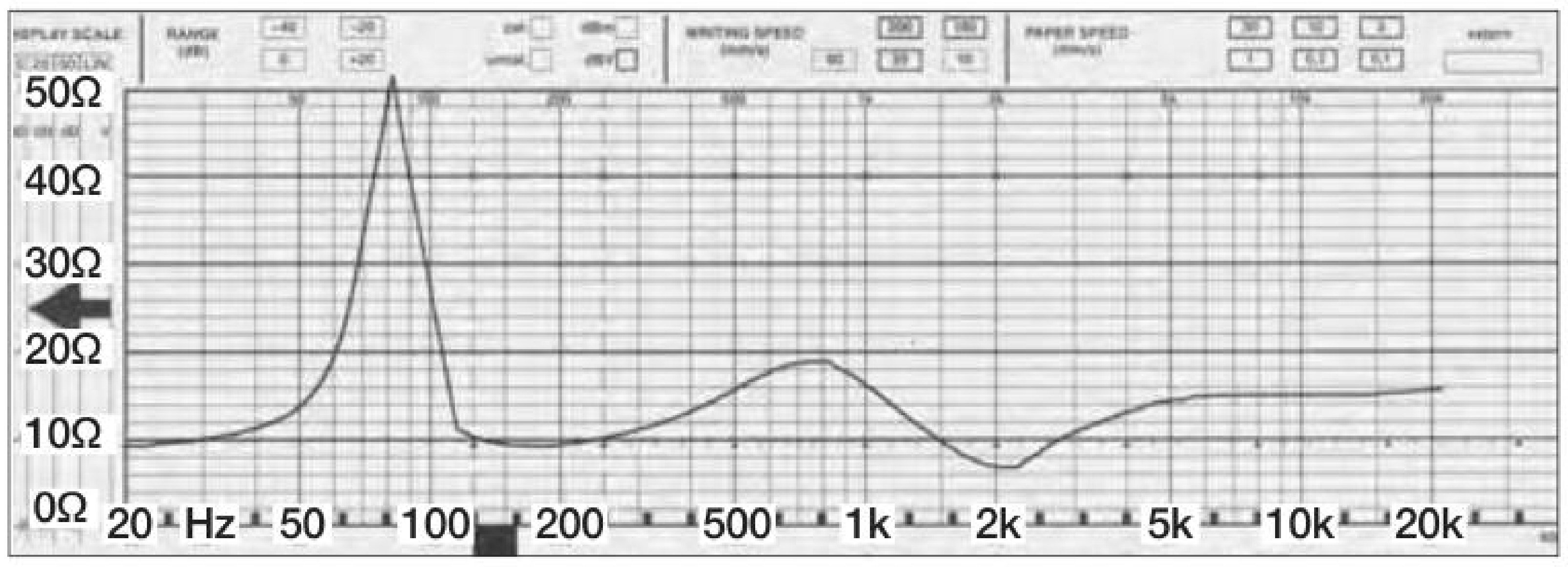


Fig.17. Impedance curve of first crossover in Fig.16 without damping network.

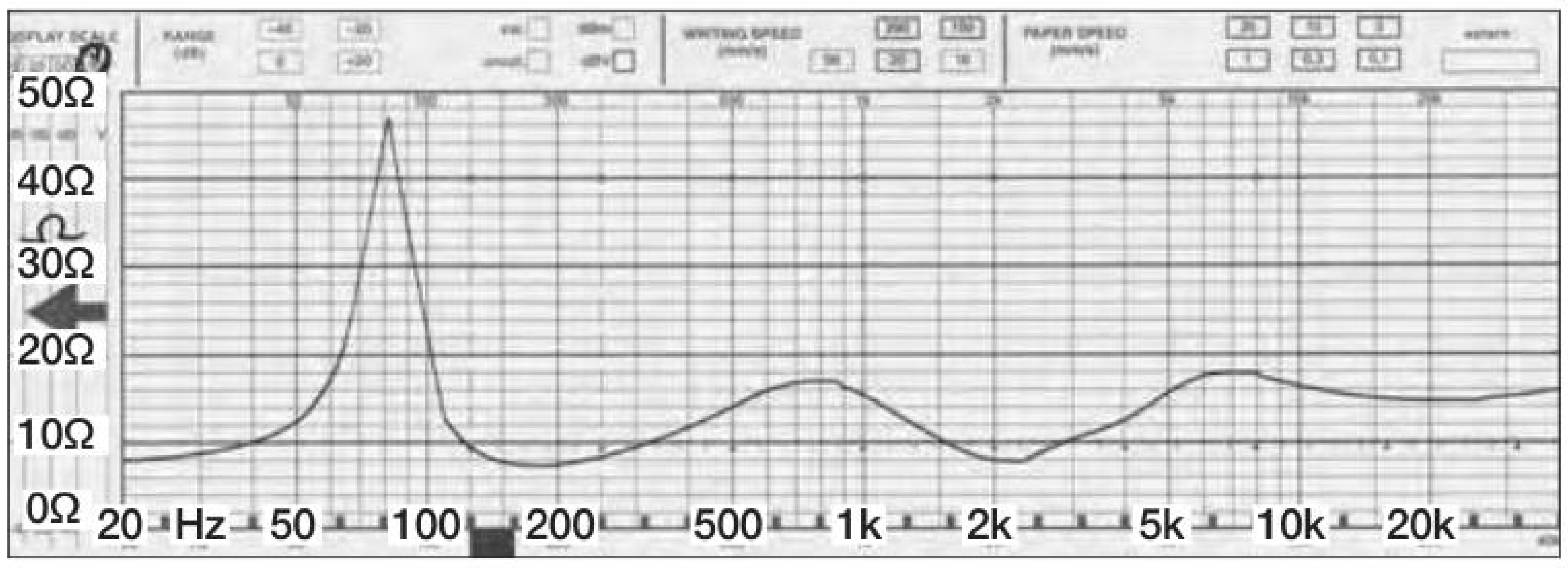


Fig.18. Impedance curve with damping network (9, C2b). The impedance curve for the final circuit in Fig.21 is the same.

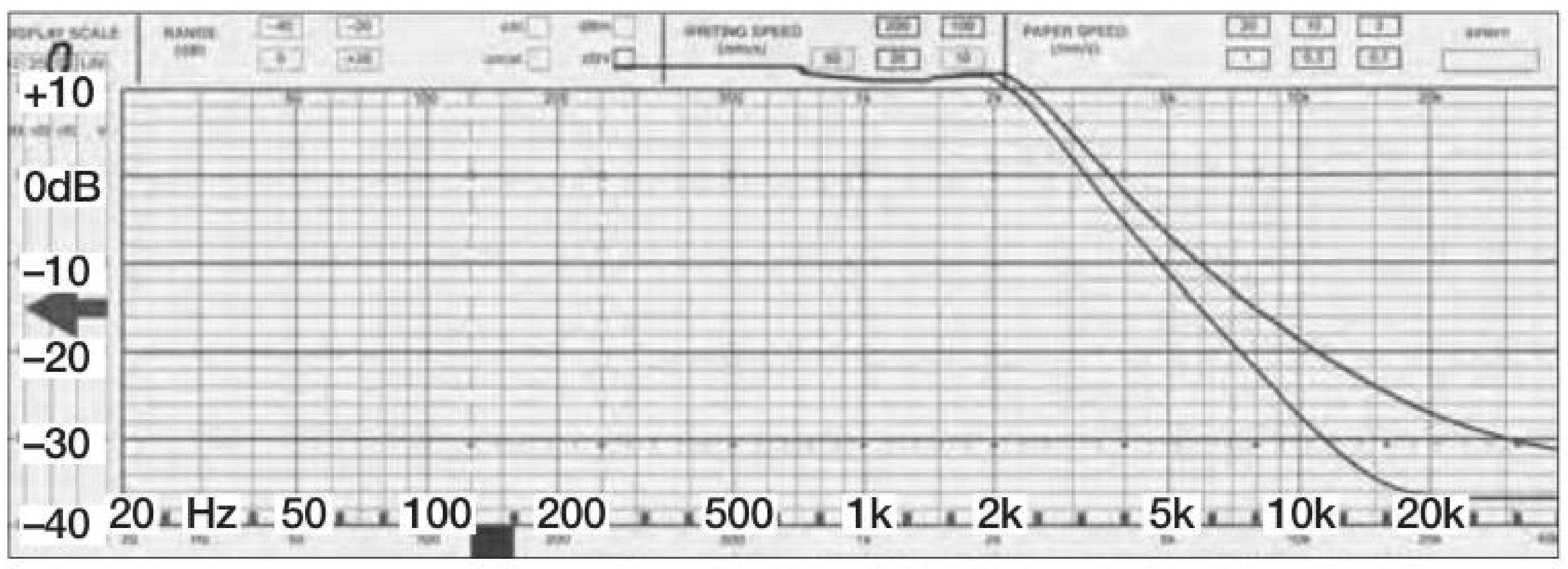


Fig.19. Improvement to the low-pass attenuation at 10kHz curve by adding C3a and C4.

minimise this. Also, the low crossover frequency made the phase shift in wavelength terms quite low; so low it cancelled out with the electrical phase shift from the filters. I didn't even have to reverse the phase of the tweeter, which is normally the case with crossovers of this topology. Listening to white noise, it is possible to identify phase notches around the crossover point by moving up and down in front of the speaker. (Do this when nobody's watching!) The white noise should 'gel' on-axis to the cabinet at a point between the two drive units. An objective test with any crossover is to deliberately connect the tweeter out of phase. A deep symmetrical null should develop (as shown later in Fig.32).

Little tweaks

Once the basic crossover has been sorted out some refinements can be made, often involving listening tests, called 'voicing'. The expensive components are the inductors (usually about £4 to £6 each) so we avoid adding more. Most tweaks just involve relatively cheap resistors and capacitors, generally adding extra refinement to the upper-mid and treble area. Some tweaks do not show up on the acoustic response curve because they are low down in the stop-band area of the crossover curves; but they can still be heard. Because of this, some electrical curves (Fig.19 and Fig.20b) are plotted on a 50dB scale.

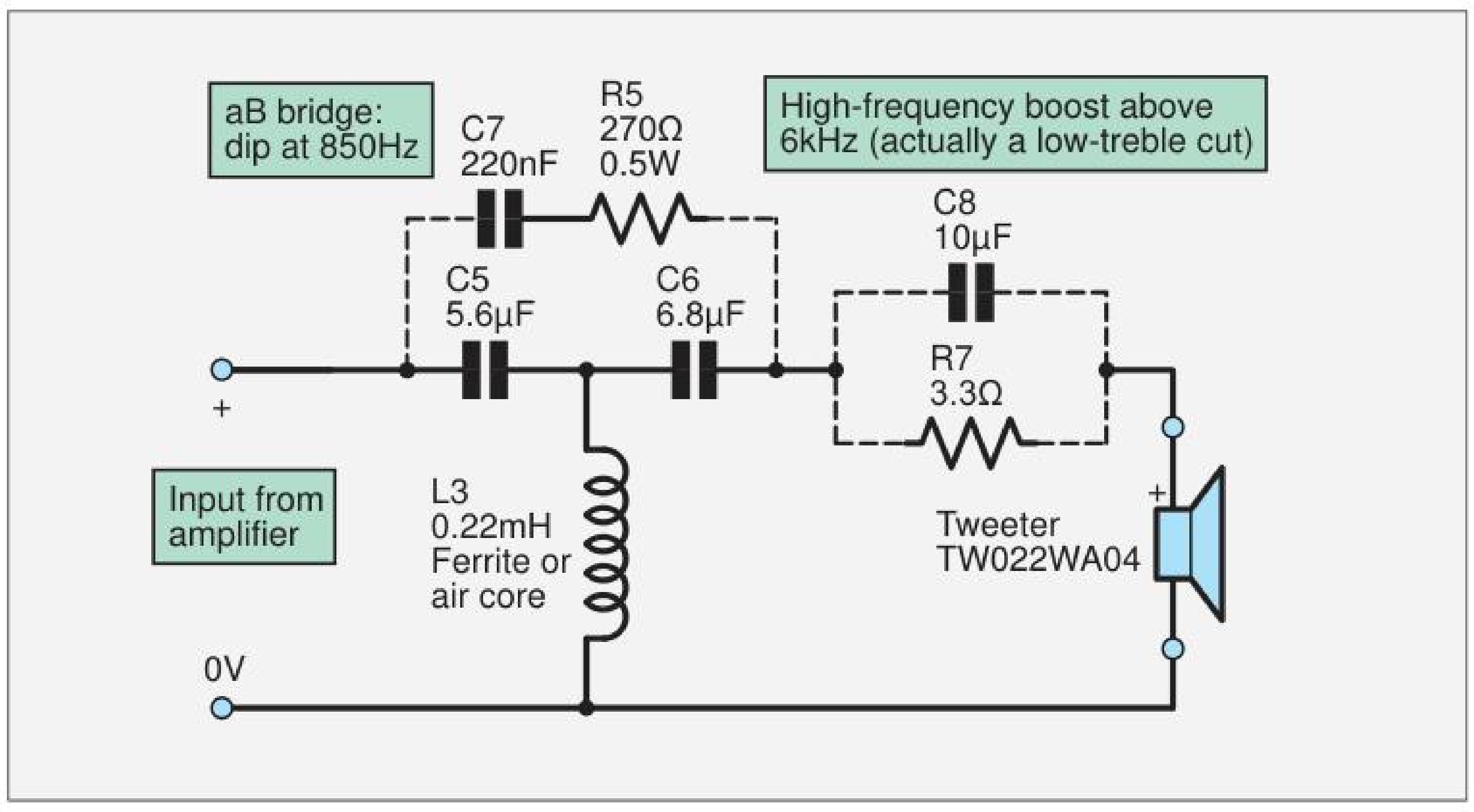

aB bridge:
dip at 850Hz
C7
220nF
R5
270Ω
0.5W
High-frequency boost above
6kHz (actually a low-treble cut)
C8
10µF
C5
5.6µF
C6
6.8µF
R7
3.3Ω
+
Input from
amplifier
L3
0.22mH
Ferrite or
air core
Tweeter
TW022WA04
+
0V

Fig.20a. Adding a tweeter anti-resonance dip using an aB network (C7, R5). Note also added a treble boost network (C8, R7).

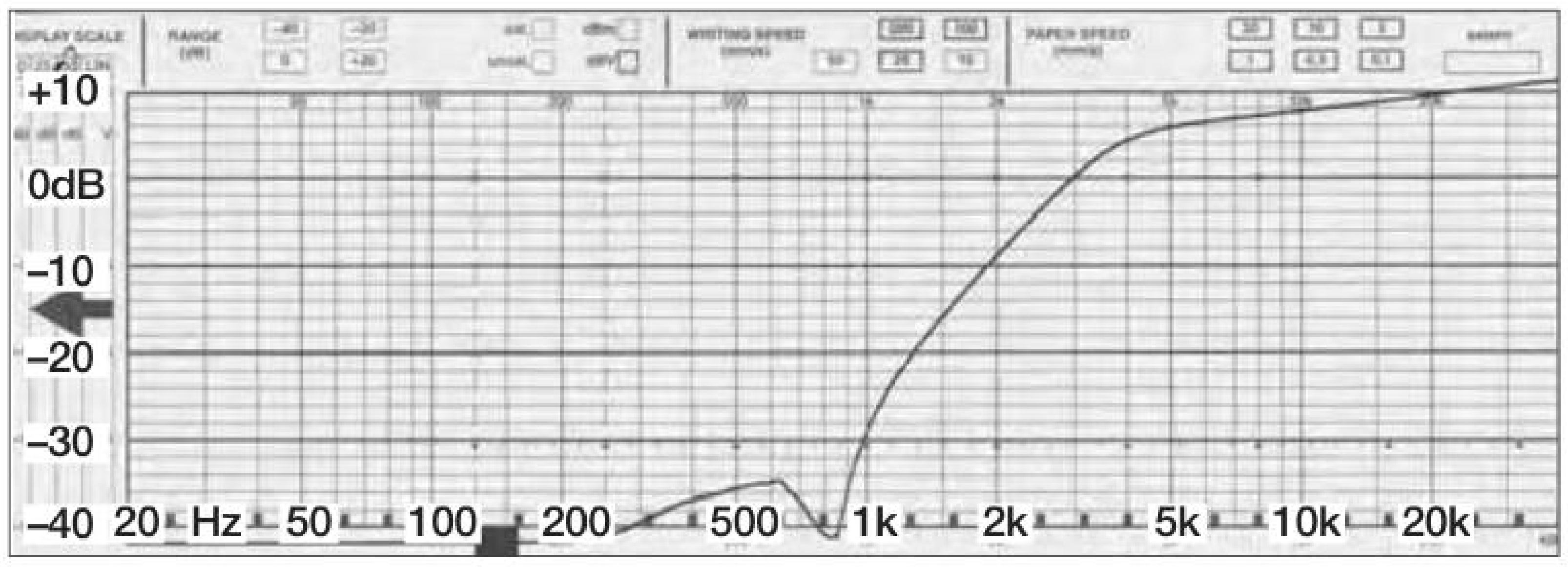


Fig.20b. Resulting tweeter anti-resonance dip using an aB network (C7, R5).

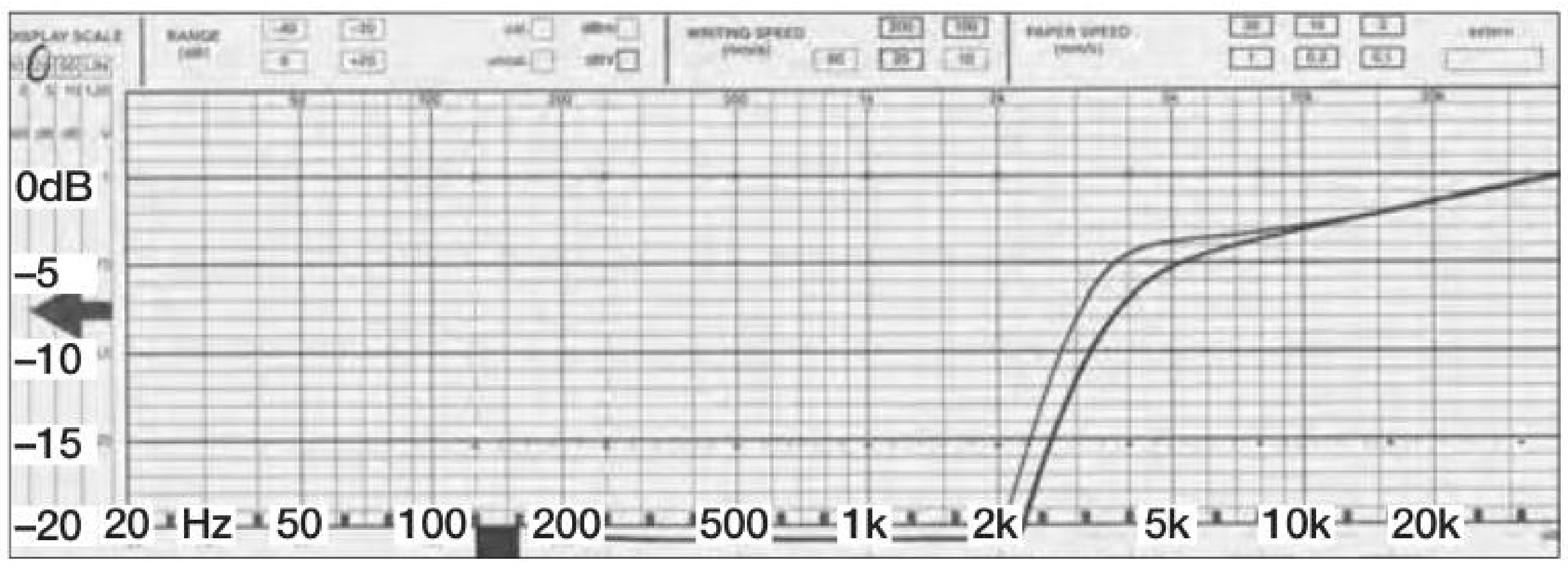


Fig.20c. Adding a treble boost network (C8, R7) cuts the low treble signal slightly.

Peak suppression

The 10kHz peak is still slightly audible on the woofer output on-axis with the tweeter disconnected. Further attenuation can be obtained by notch-tuning the final inductor with a parallel capacitor C3a. A capacitor C4 in parallel with the woofer also helps. The difference to the low-pass electrical curve is shown in Fig.19. It just reduces the 10kHz by about 6dB. Since there is now no treble radiated from the woofer, the overall treble response sounds a bit smoother because there are no high-frequency cancellation notches.

Softer treble

The crossover as it stands may sound a little 'hard' or 'mid-rangey' to older listeners (50+) with a degree of age-related high-frequency loss or presbycusis, which I have. This can be ameliorated with a bit of treble lift on the amplifier or by inserting the network consisting of C8 and R7 in series with the tweeter after the high-pass filter. This decreases output in the crossover region a little and boosts it a few dB at 10kHz. This gentle rise also compensates for reducing the high-frequency output off-axis from the tweeter. C6 is also reduced to the more common value of 6.8µF because of the increased impedance.

Tweeter aB section

Here's another subtle tweak, one considered by KEF and others to smooth the high-pass curve at the lower end. It suppresses the tweeter's 850Hz resonant peak using an acoustic-Butterworth bridge consisting of C7 and R5. I could only hear it on white noise with the woofer disconnected, but it costs less

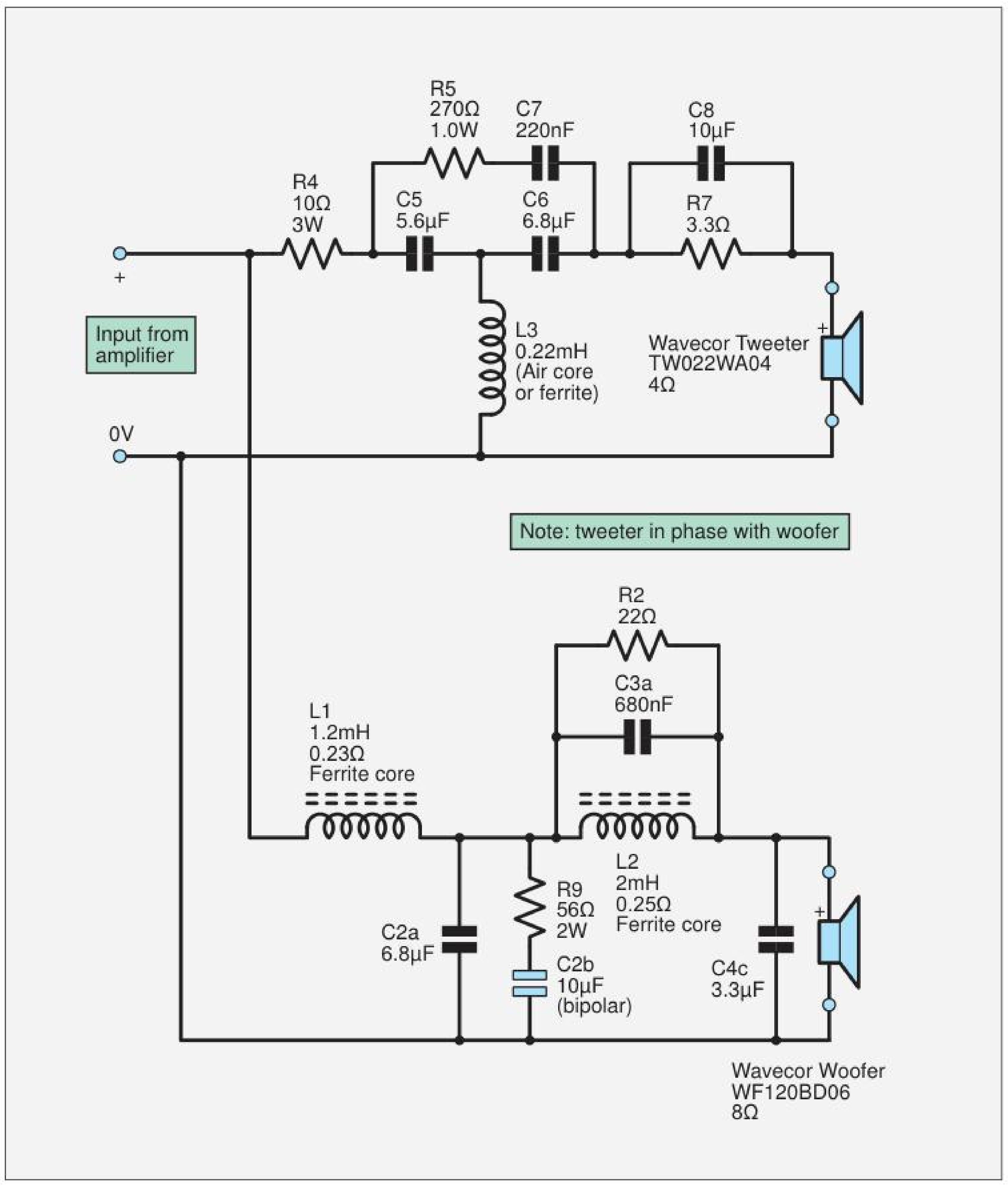
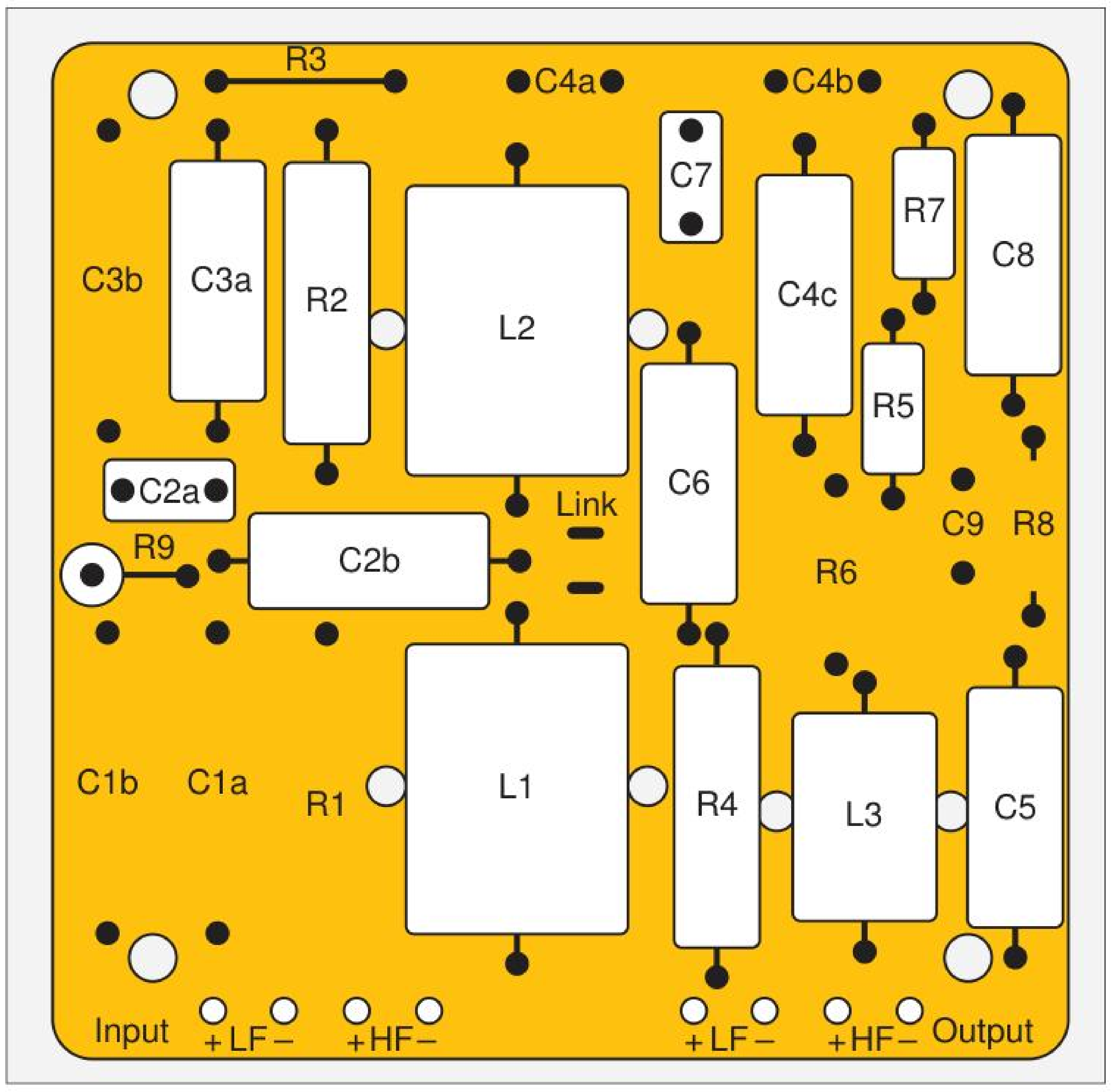

R5
270Ω
1.0W
C7
220nF
C8
10µF
R4
10Ω
3W
C5
5.6µF
C6
6.8µF
R7
3.3Ω
+
Input from
amplifier
L3
0.22mH
(Air core
or ferrite)
Wavecor Tweeter
TW022WA04
4Ω
+
0V
Note: tweeter in phase with woofer
R2
22Ω
C3a
680nF
L1
1.2mH
0.23Ω
Ferrite core
L2
2mH
0.25Ω
Ferrite core
R9
56Ω
2W
C2a
6.8µF
C2b
10µF
(bipolar)
C4c
3.3µF
+
Wavecor Woofer
WF120BD06
8Ω

Fig.21. Final crossover circuit with all the tweaks. These did not alter the impedance curve.



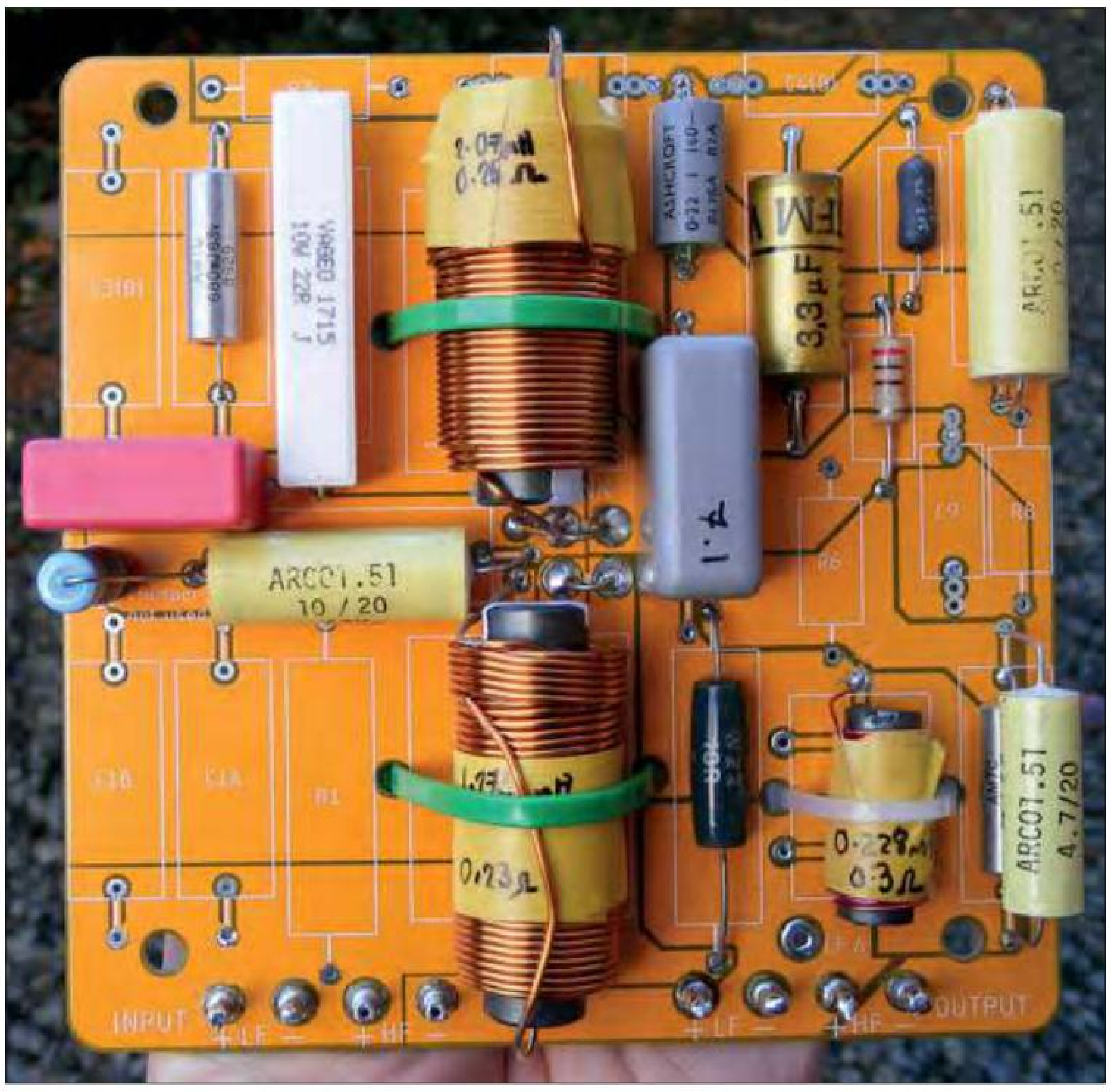


Fig.22. (above left) Component overlay for *PE Mini-Monitor* crossover on the *Universal Passive Crossover* PCB. (Note: the tweeter phase marking on the PCB is incorrect and should be reversed – ie, bottom right, swap '+' and '–' either side of 'HF' output. (It is correct for the LS3/5A version, which has an inverted tweeter connection, but the tweeter for the *PE Mini-Monitor* is wired in-phase.) Also note the two central links need to be soldered to the board. Fig.23a. (above right) shows the completed *PE Mini-Monitor* crossover (do include the R3 wire link!). 23b (below) Note the vertical mounting for R9.

than £1, so it's worth it. The modified circuit and resulting curve are shown in Fig.20a and Fig.20b. This feature is more important for third-order high-pass filters where the tweeter has no inductor across it or other means to damp the resonance. There is an option for the tweeter to have ferrofluid in its magnetic gap, which provides high damping and power handling. I don't like it, because the oil dries/thickens after a few years and sounds bad at low levels.

Final crossover circuit

This is shown in Fig.21. There are a lot of 'Rs and Cs', but the minimum of inductors. I asked Volt Loudspeakers in Devon to wind these and was gratified to find their DC resistance was lower than they originally specified and their tolerance was within a few percent.

Final frequency response

Building the crossover on the *Universal PCB* makes a refreshing change from the normal microscopic components of today. Fig.22 shows the overlay and Fig.23 a photo of the completed unit. Plated-through-holes with leaded components gives maximum strength, ideal for a crossover being shaken about inside a loudspeaker. It's a good idea to wind a piece of insulation tape round the inductor where the termination wire is clamped under the cable tie. This prevents a shorted turn forming from abrasion of the enamel arising from vibration. Hot glue, foam and electronic grade (acid-free) silicone can reduce possible buzzes.

Parts list

Notes

- The following list is *per crossover* – two sets needed for a pair!
- All components are ±5% tolerance or better. The most important thing is that the components on left and right crossovers are matched for good stereo imaging. So if, for example, you have two 10µF capacitors that are 10% too high, put them in the same position on each board.
- 'WW' means wire-wound.

Resistors

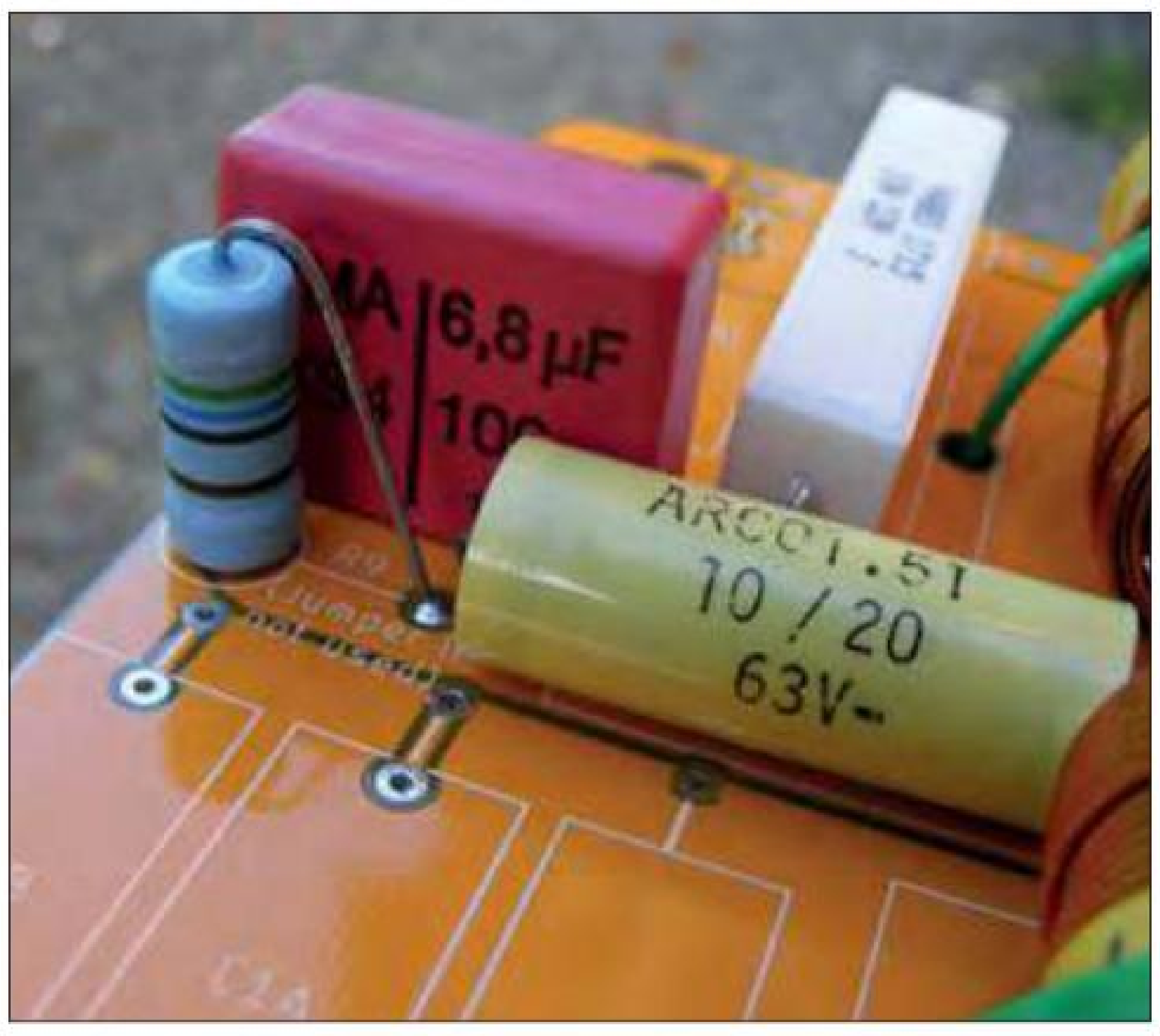
R2	22Ω 6W WW
R4	5.6 to 12Ω 2.5W WW, select value according to taste to set tweeter level
R5	270Ω 0.5W carbon-film
R7	3.3Ω 2.5W WW
R9	56Ω 2W metal-oxide or WW

Capacitors

Note the *Universal Passive Crossover* PCB has multiple holes, so constructos can use either radial or axial types

For plastic film capacitors, use polypropylene (MKP), polycarbonate (MKC) or polyester (MKT) – in that order of preference – rate at 63V minimum.

C2a, C6	6.8µF
C2b, C8	10µF
C3a	680nF
C4c	3.3µF
C5	5.6µF from Blue Aran – part no. CVMPC25560 (or parallel combination: 4.7µF and 820nF)
C7	220nF



Inductors

L1	1.2mH 0.23Ω 1mm WW on 12.5 x 50mm core (Neosid F6 manganese ferrite rod part number 36-702-26). Volt part number F002.
L2	2mH 0.25Ω same construction as L1. Volt part number F020
L3	0.22mH 0.3Ω 9mm or 12mm x 25mm long ferrite. Alternatively, use an air-core type, 0.21mH 0.4Ω (Volt type 72).

L1, L2, L3 (ferrite core) coils made for *PE* by Volt Loudspeakers are available as a complete set, with the resistors and capacitors from the *PE* shop – Part WAVXO.

Miscellaneous

Cable ties 3mm by 100mm, 3off
Universal Passive Crossover PCB from the *PE Shop* – Part UPC0320
2mm turret tags, 12 off
22swg tinned copper wire for links, 100mm
19mm-wide PVC insulation tape, 150mm

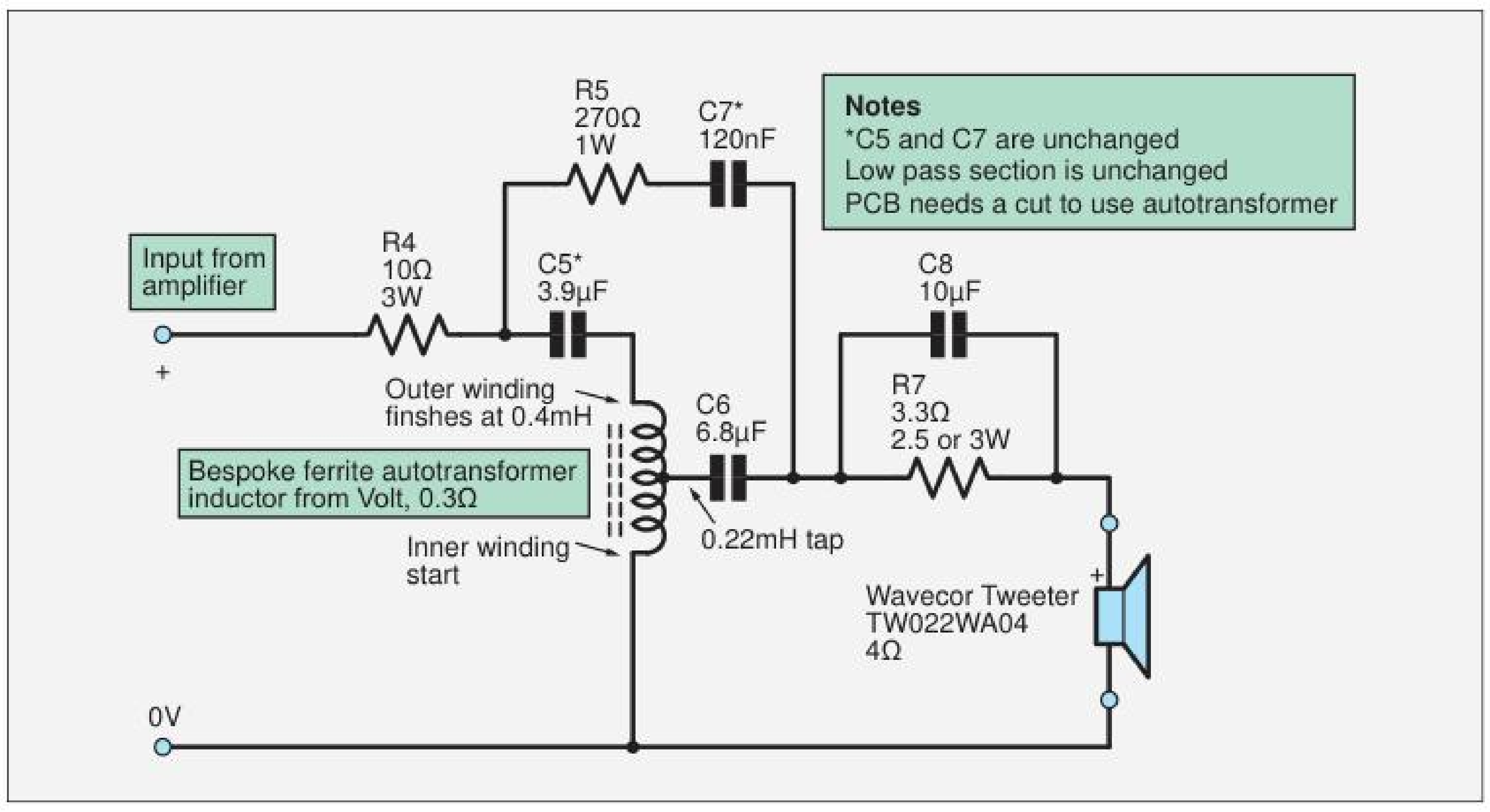

R5
270Ω
1W
C7*
120nF
Notes
*C5 and C7 are unchanged
Low pass section is unchanged
PCB needs a cut to use autotransformer
Input from amplifier
+
R4
10Ω
3W
C5*
3.9µF
C8
10µF
Outer winding finshes at 0.4mH
C6
6.8µF
R7
3.3Ω
2.5 or 3W
Bespoke ferrite autotransformer inductor from Volt, 0.3Ω
0.22mH tap
Inner winding start
Wavecor Tweeter
TW022WA04
4Ω
0V

Fig.24. New high-pass section using auto transformer.

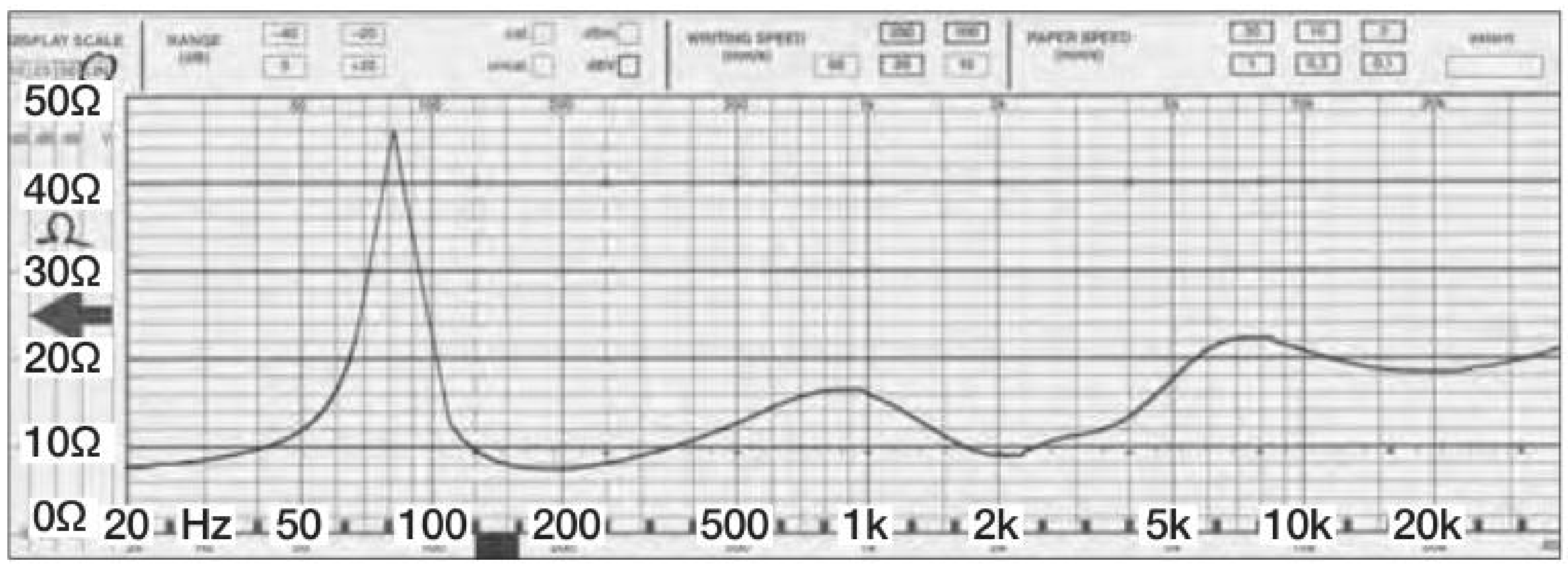


Fig.25. Impedance of the auto-transformer version, significantly higher in the top-end.

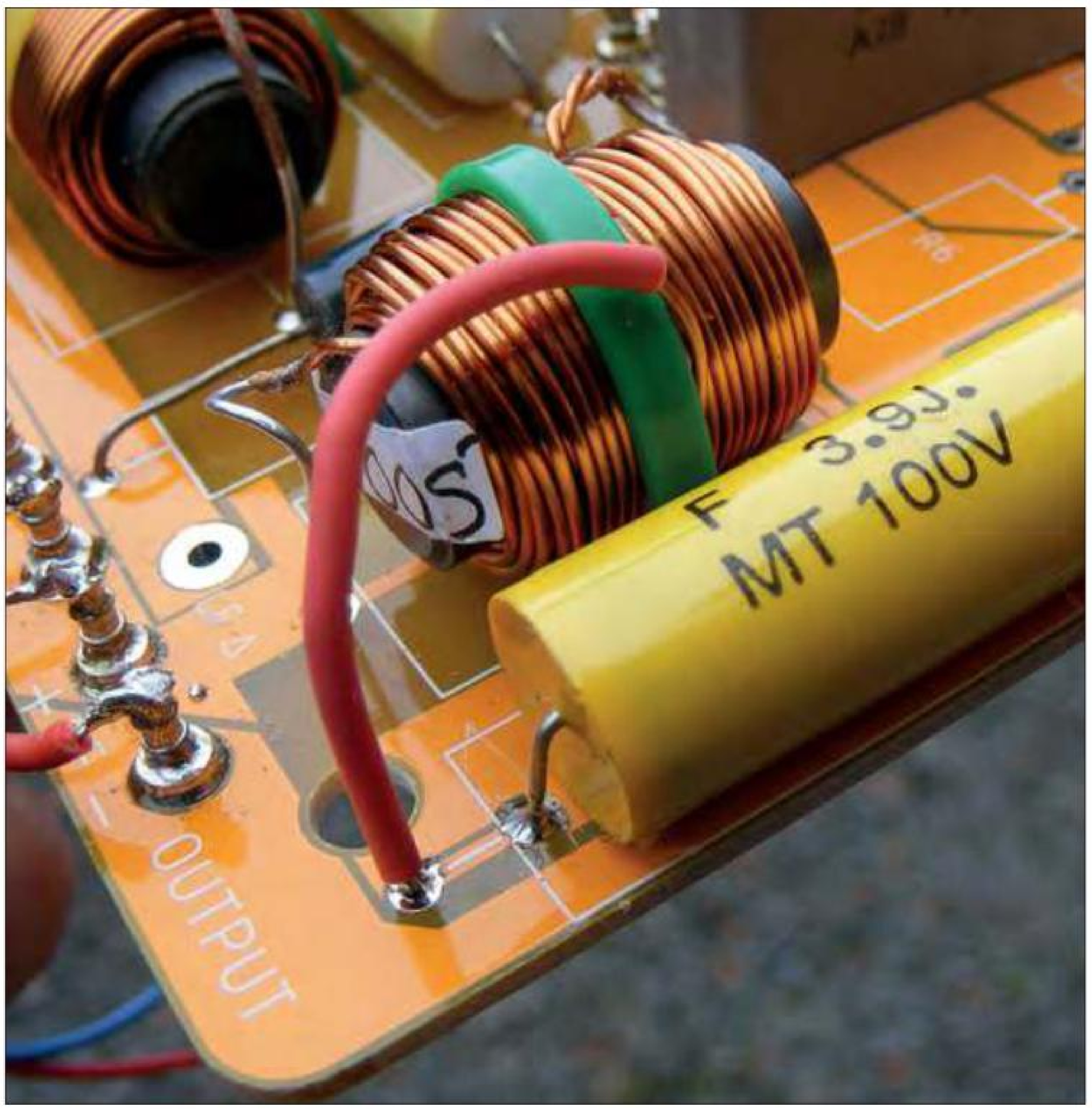

F 3.9J.
MT 100V
OUTPUT

Fig.27. The wire from the outer winding of the tapped coil is fed into the isolated track using the spare capacitor hole.

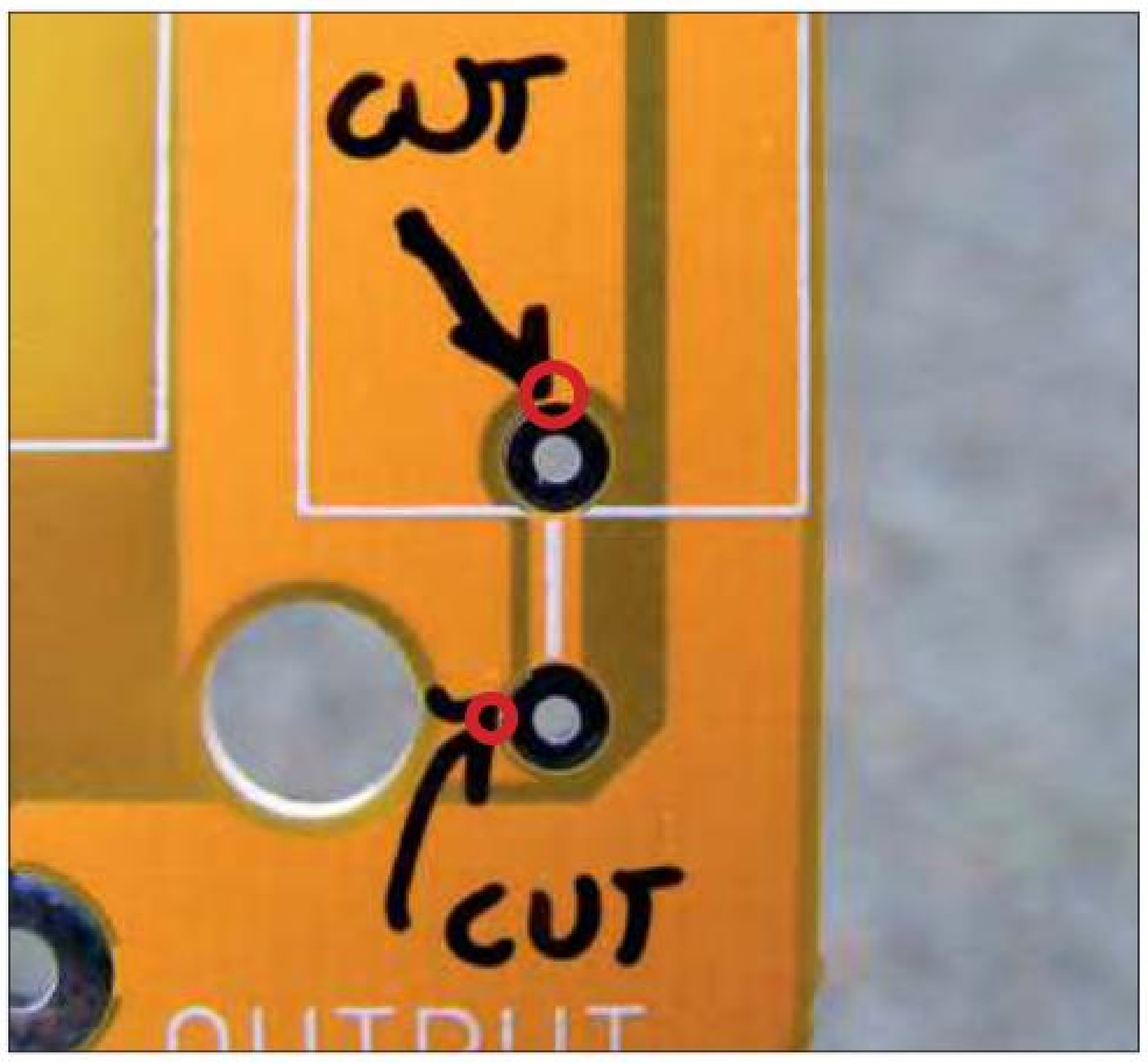


Fig.26. To use the auto transformer the PCB needs the track to be isolated by cutting as shown. This will be fully covered in a future article.

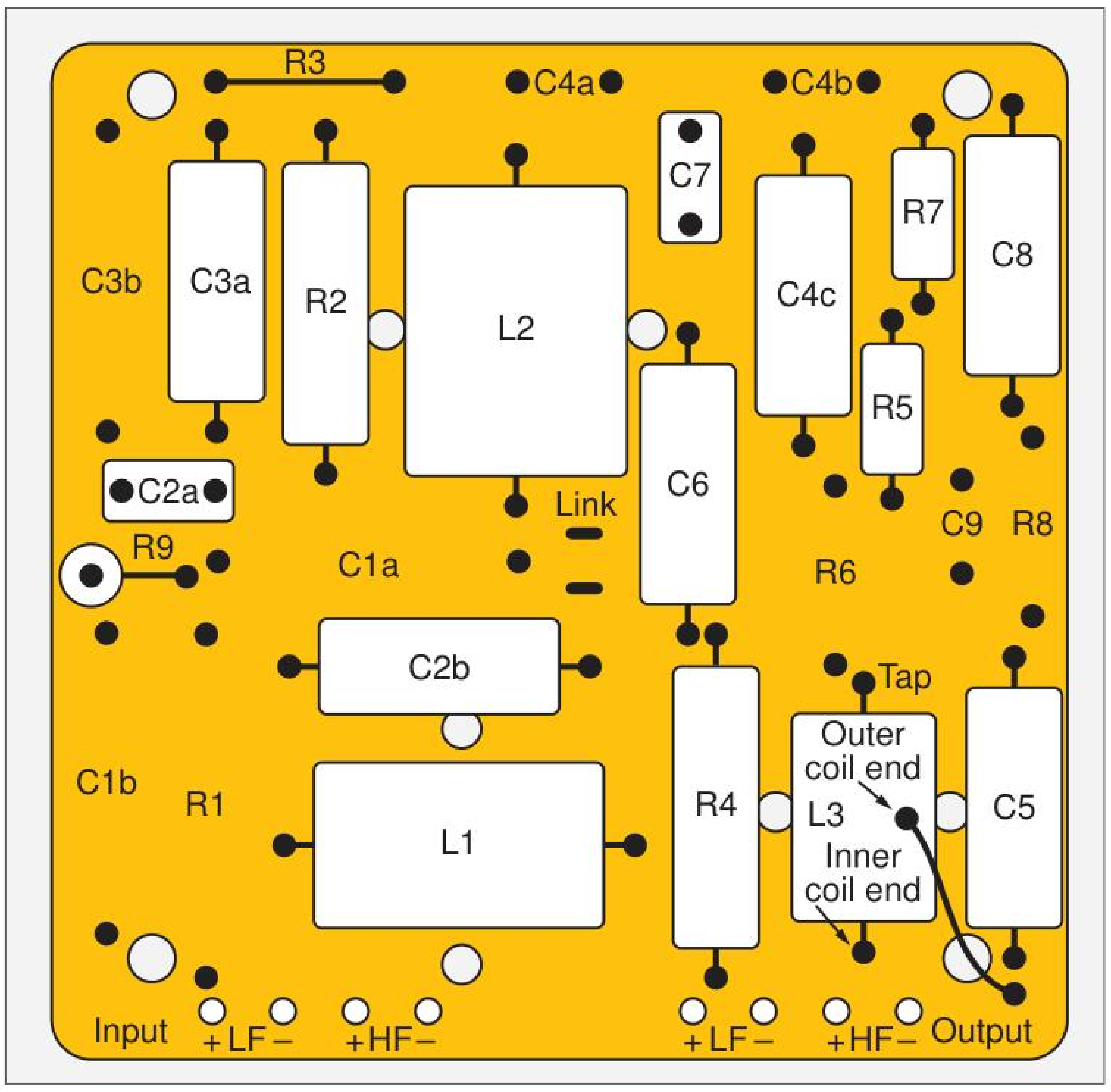
Autotransformer – future upgrade

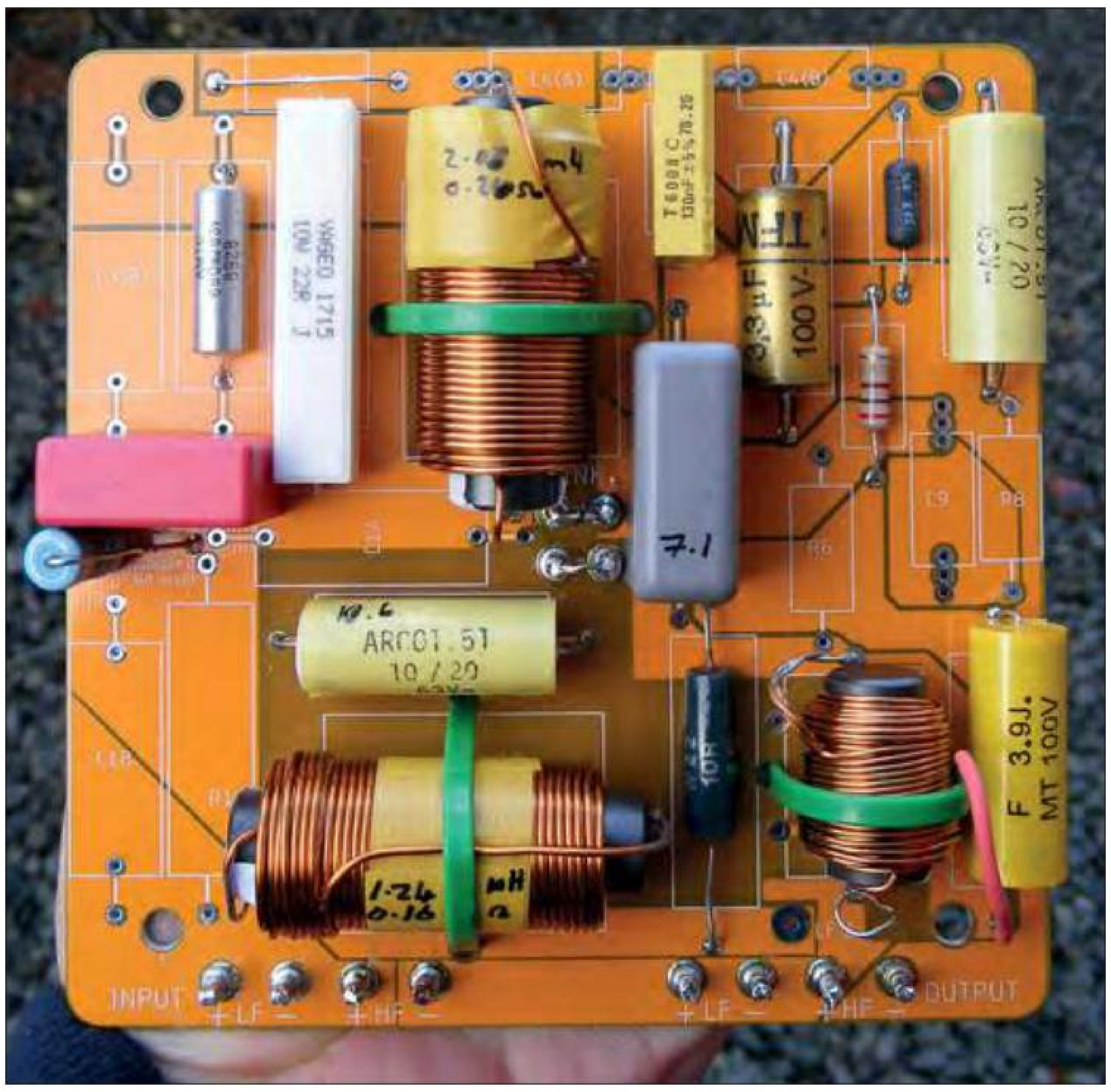
I have commissioned Volt to wind some tapped inductors to implement the higher-impedance approach used in the LS3/5A crossover. This helps avoid the loss of efficiency in matching a 4Ω tweeter to an 8Ω woofer. These are 0.4mH tapped at 0.22mH. The new high-pass section is shown in Fig.24 and because the input impedance is higher, the first capacitor C5 has to be reduced to 3.9µF (**mouser.co.uk** part ECW-FD2W395K) The aB capacitor value C7 also has to be altered to 120nF (available from Tayda). The response curve is the same, however the impedance curve shown in Fig.25 is now 4Ω higher in the treble region, reducing amplifier distortion. There is an important construction detail to take note of when this part becomes available: the PCB will need to be cut, see Fig.26. The autotransformer wiring is shown in Fig.27: feeding the coil from one of the spare capacitor (C5) pads. This will be fully covered in a later article.

Final edit

One of the 'perils' of CAD PCB design is that it is easy to change and modify

Fig.28. (below-left) Overlay for the new board layout with the low frequency coils (L1 and L2) at right angles. Also note that the positions of capacitors C1a and C2b have changed. The (Fig.22) note about the tweeter output also applies here.
Fig.29. (below-right) Photo of the new board with the auto-transformer.


R3
C4a
C4b
C7
R7
C8
C3b
C3a
R2
L2
C4c
R5
C6
C2a
Link
C9
R8
R9
C1a
R6
C2b
Tap
Outer coil end
C1b
R1
L1
R4
L3
C5
Inner coil end
Input
+LF–
+HF–
+LF–
+HF–
Output


INPUT
+LF
+HF
+LF
+HF
OUTPUT

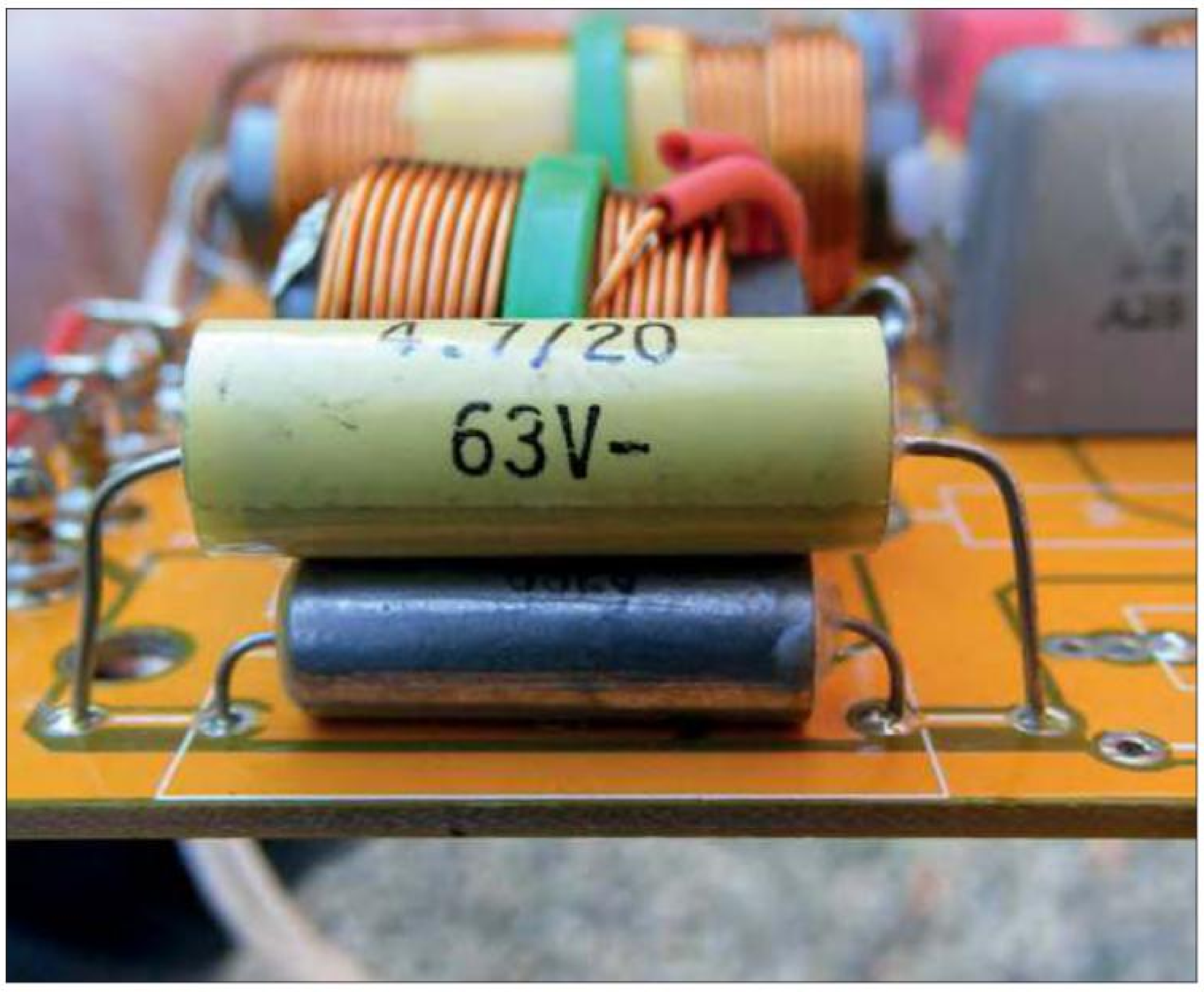


Fig.30. The multi-pad connections can be used to parallel components for tolerance adjustment or making up values. In this case C5 (5.6µF) has been made up of a 'high' 4.7µF and a 680nF.

things very easily. I felt uneasy with the two inductors (L1 and L2) in the low-frequency section being in-line because it says in every text book 'don't do it, put them at right angles'. This advice mainly applies at high frequencies, not the low frequencies involved here. Also, the overall magnetic circuit path length was quite long. The end result being I could measure no difference, but I changed the board anyway. The only other things that had to change were the positions of C1a and C2b. Fig.28 shows the overlay of the revised board and Fig.29 a photo of the completed board with autotransformer L3 installed.

CRICKLEWOOD ELECTRONICS
Established 1981

Your best bet since MAPLIN
Chock-a-Block with Stock

Visit: www.cricklewoodelectronics.com
Or phone our friendly knowledgeable staff on 020 8452 0161
Components • Audio • Video • Connectors • Cables
Arduino • Test Equipment etc, etc

Visit our Shop, Call or Buy online at:
www.cricklewoodelectronics.com
020 8452 0161
Visit our shop at:
40-42 Cricklewood Broadway
London NW2 3ET

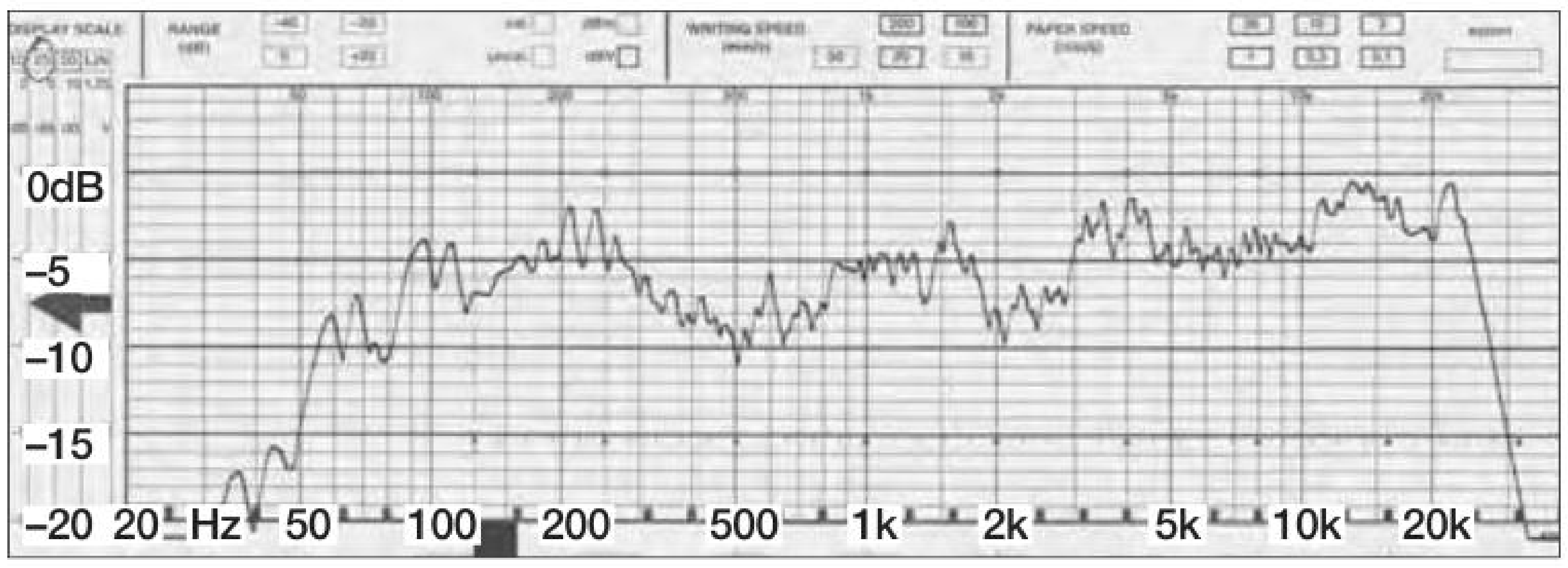


Fig.31. Final frequency response using the tweaked crossover on a two foot open stand in a semi-anechoic living room. A bit of BBC style 'voicing' to compensate for the small size has been applied. Notice the slight bass hump to compensate for the lack of deep bass. There is a slight dip at the crossover to compensate for the sound being too 'forward' for close monitoring. Also, there is a high-frequency gentle rise on axis to compensate for the reduction in dispersion off axis. This response is not as smooth as the original curve (Fig.16b, see last month) but in fact, it 'sounds' smoother.

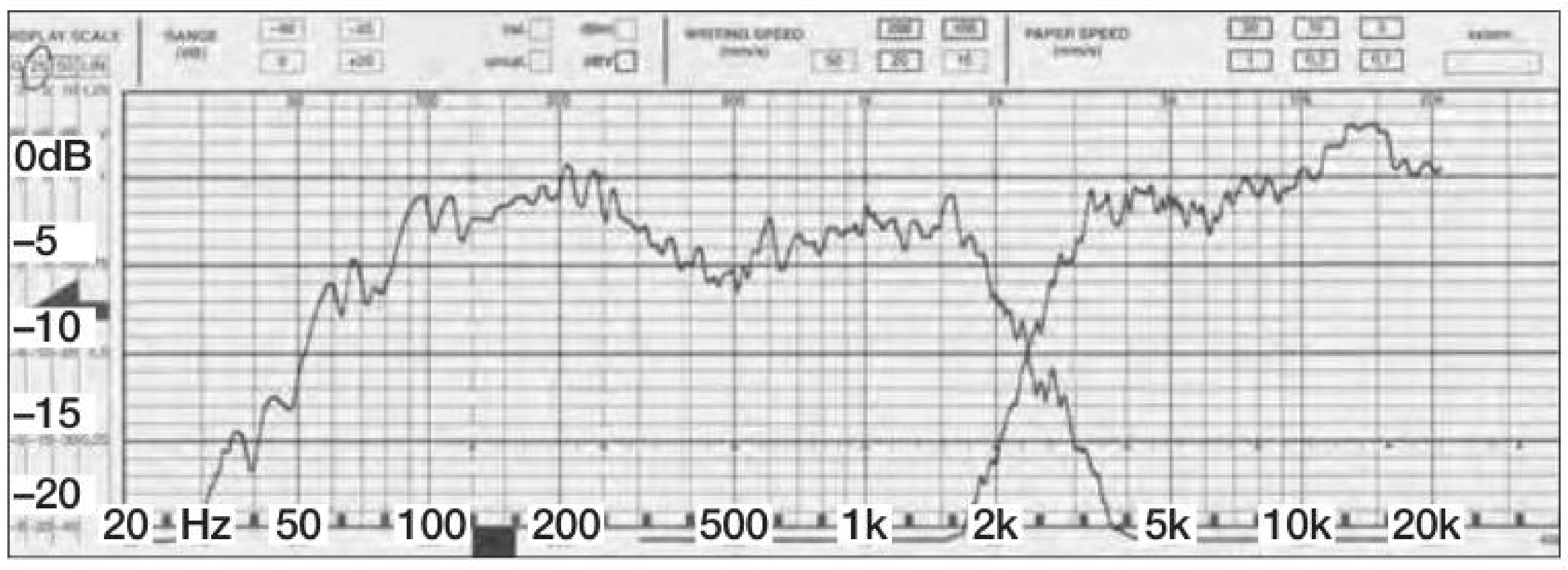


Fig.32. Crossover curve showing the woofer and tweeter outputs separately. The woofer slope is around –15dB per octave and the tweeter nearer –24dB/oct. The slopes are monotonic, that is they go down continuously without jumping up again. The different rates of attenuation are desirable because of the difference in directivity for each driver at the crossover point. (The woofer is beginning to 'beam' and the tweeter is almost omni-directional).

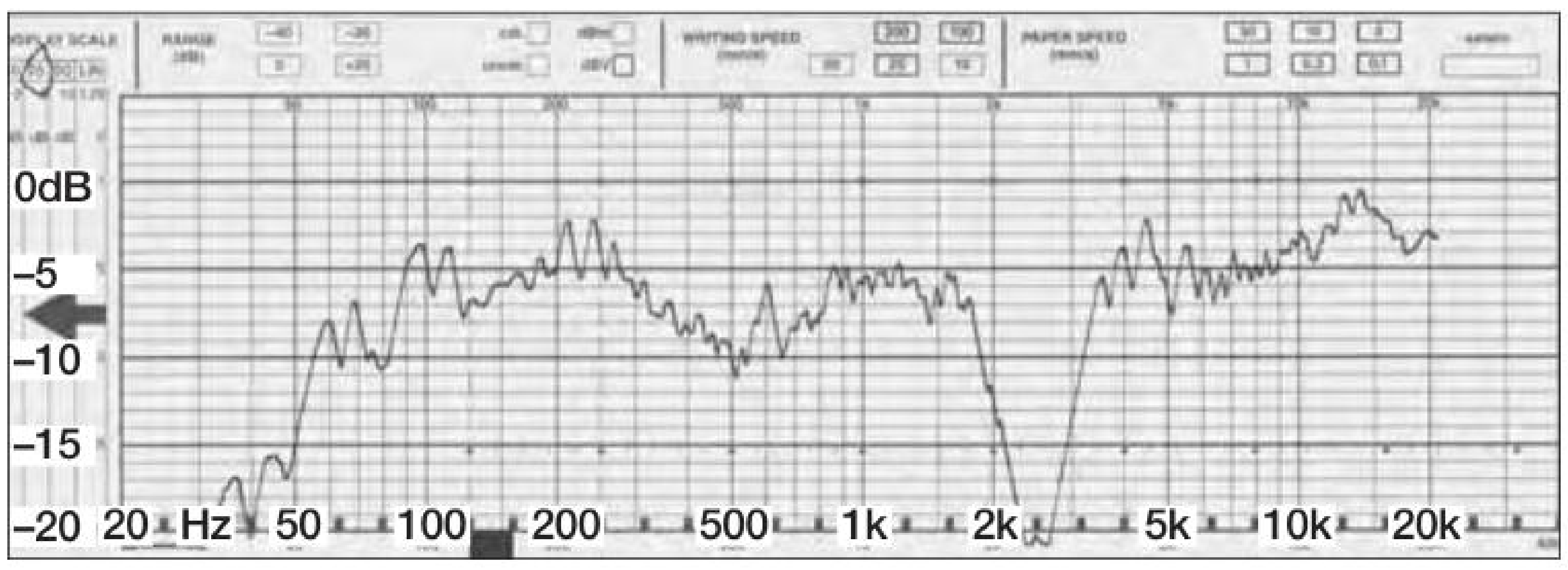


Fig.33. The response with the tweeter connected anti-phase relative to the woofer. It is deep and symmetrical, indicative of good phase control through the crossover region. If you can't hear this I suggest you find an audiologist.

Another interesting component insertion technique is optimisation of capacitor values by paralleling components: mounting them one on top of the other, as shown in Fig.30. This is particularly handy if either you need to match a pair of values on – for example a pair of crossover boards – or, if you want create an unobtainable component that lies between two E-series values; for example, 4µF is within 0.5% of 3.3µF + 680nF.

Final listening

The final frequency response, incorporating the tweaks is shown in Fig.31. The separate curves for both drivers are shown in Fig.32. The anti-phase connection is shown in Fig.33. I may be biased, but I have just had a good session with the *PE Mini-Monitors* and I think they give superb clarity and spatial imagery for the price (around £300). I was listening to Goldfrapp and it brought tears to my eyes, a true indicator of effective audio design. (Here I am even more biased – Alison Goldfrapp has bought my theremins!)

Make it with Micromite

Phil Boyce – hands on with the mighty PIC-powered, BASIC microcontroller

Part 14: Using Bluetooth to set your Micromite free

So far in this series, if you want to interact with your *Micromite Keyring Computer* (*MKC*) you have to physically connect it (and the *Development Module* (*DM*)) to your computer with a USB cable. Then, you run a terminal application such as TeraTerm on your computer. The terminal application acts purely as a 'console' which provides the Micromite with a screen (for output), and a keyboard (for input). You can think of this set-up as the Micromite being 'USB tethered' to a screen and keyboard – see Fig.1. Note that the physical length of the USB cable, which for technical reasons is usually no more than one metre, determines the maximum distance between the Micromite and computer.

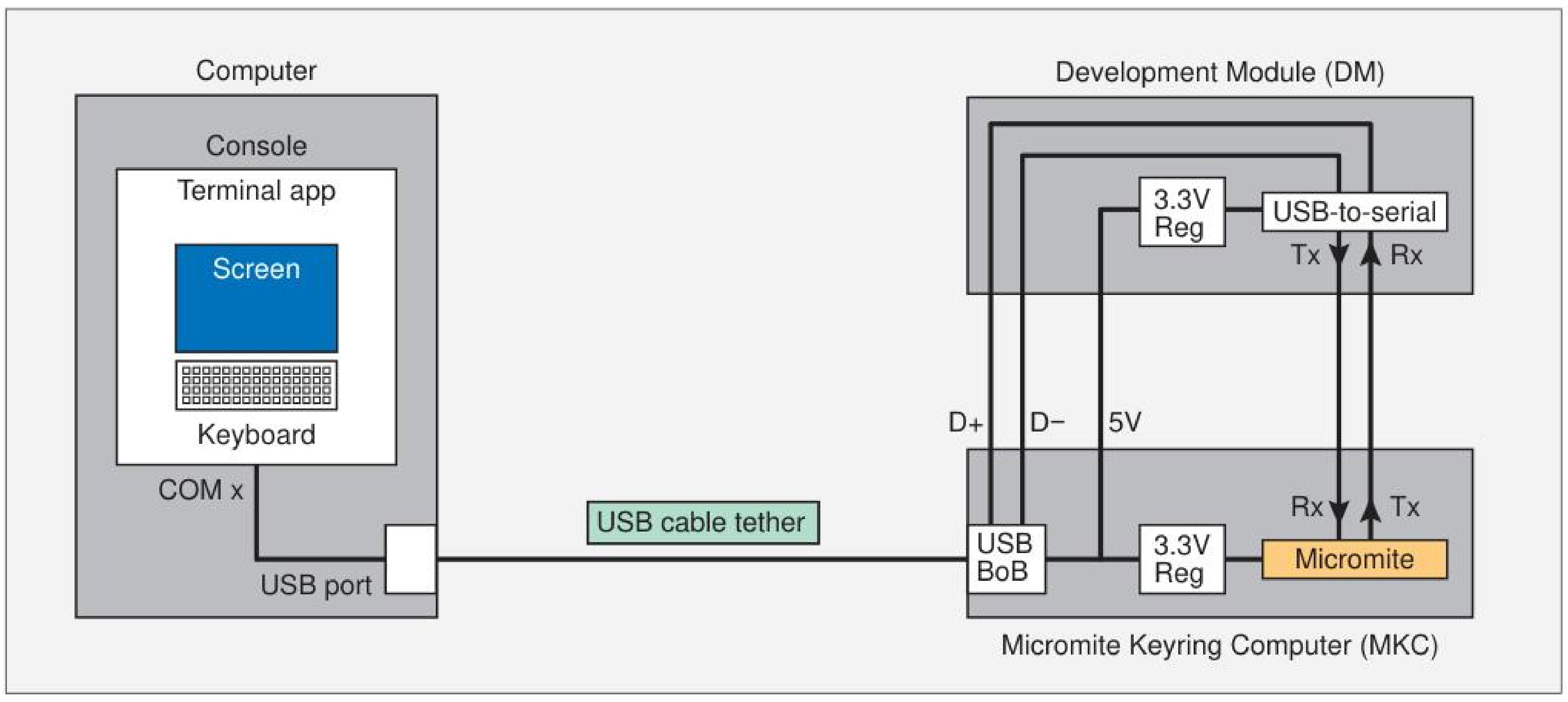

Computer
Console
Terminal app
Screen
Keyboard
COM x
USB port
USB cable tether
Development Module (DM)
3.3V Reg
USB-to-serial
Tx
Rx
D+
D−
5V
Rx
Tx
USB BoB
3.3V Reg
Micromite
Micromite Keyring Computer (MKC)

Fig.1. Using a USB cable to connect your *MKC/DM* to your computer results in them being tethered together. Distance apart is limited by length of USB cable – typically no more than 1m.

This month, we are going to show you how to 'set your Micromite free' by replacing the USB physical link with a wireless link. Using Bluetooth technology to create this wireless link, it becomes possible to separate your Micromite and computer from each other by up to 10m.

First though, why would you want to use a wireless link instead of having the Micromite tethered to within a metre of your computer? Well, a wireless link opens up all kinds of remote-control projects and capabilities; for example:

- Several sensors could be connected to a Micromite, and assembled into a waterproof enclosure that you locate outdoors. Sensor data can then be formatted in your Micromite code (with `PRINT` commands and Escape codes – see *Part 8*, September 2019) for display on your computer screen, which is located conveniently indoors.
- A battery-powered robot-buggy could be created that incorporates an on-board Micromite. A Bluetooth link would allow the robot to roam freely, totally untethered; yet it can be fully controlled from your computer's keyboard. (A Bluetooth-controlled robot-buggy will be a future project).
- You could modify any of your existing projects to enable them to be accessed remotely. For example, adding Bluetooth to last month's *Mood Light* project means that it can remain where you want it to be positioned, while customising the code from the comfort of your 'remote' computer.

These examples have probably inspired you with some ideas for wireless interaction with your Micromite. Let's get started by understanding the design of this Bluetooth link.

Design concept

In Fig.1, you can see the familiar method we have used so far in this series to connect the *MKC* to the computer's USB port via the USB-to-serial module inside the *DM*. Electrically, the USB-to-serial module deals with the complex USB

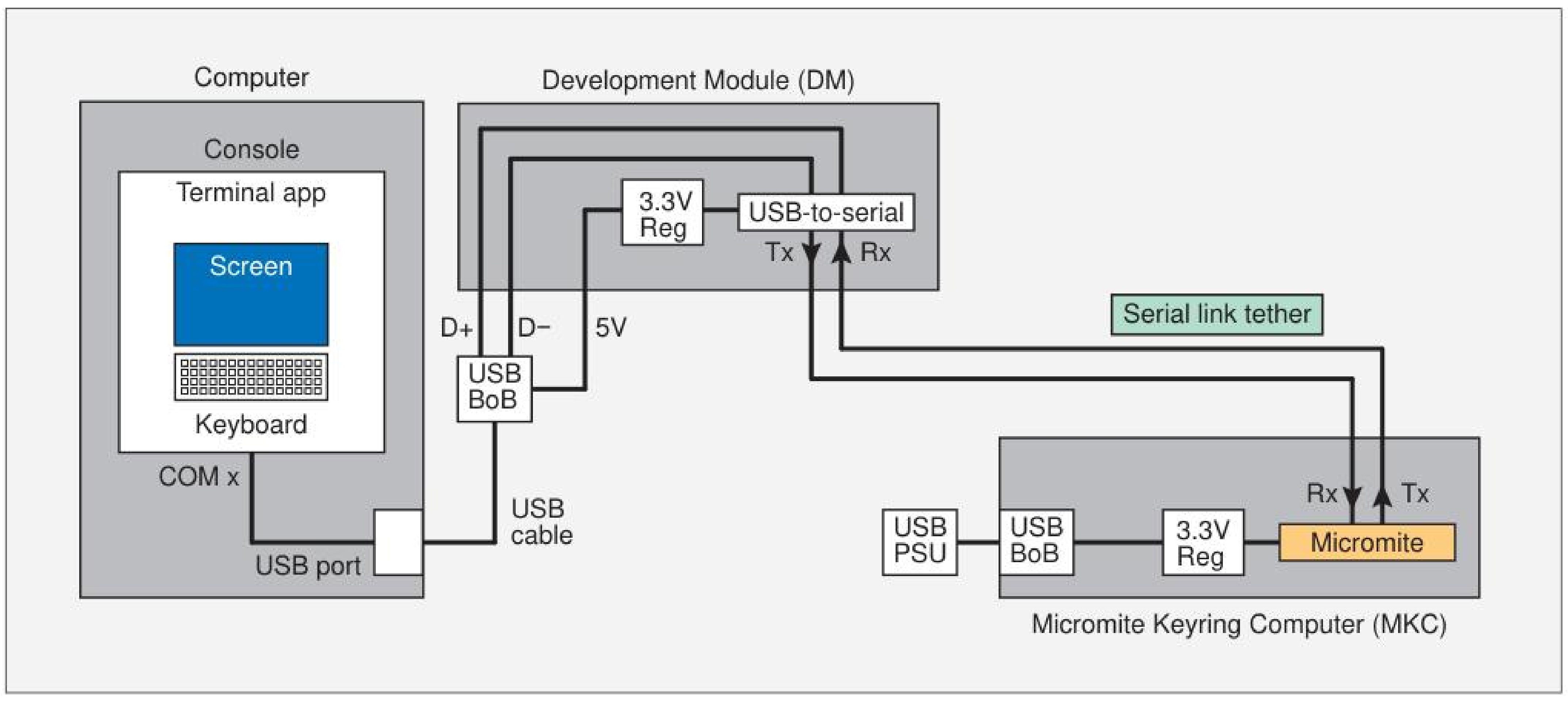

Computer
Development Module (DM)
Console
Terminal app
Screen
Keyboard
COM x
USB port
3.3V Reg
USB-to-serial
Tx
Rx
D+
D−
5V
USB BoB
USB cable
Serial link tether
Rx
Tx
USB PSU
USB BoB
3.3V Reg
Micromite
Micromite Keyring Computer (MKC)

Fig.2. Rearranging the elements in Fig.1 results in a serial link tether. Theoretically, this means the computer and Micromite can now be separated by more than 1m.

Questions? Please email Phil at:
contactus@micromite.org

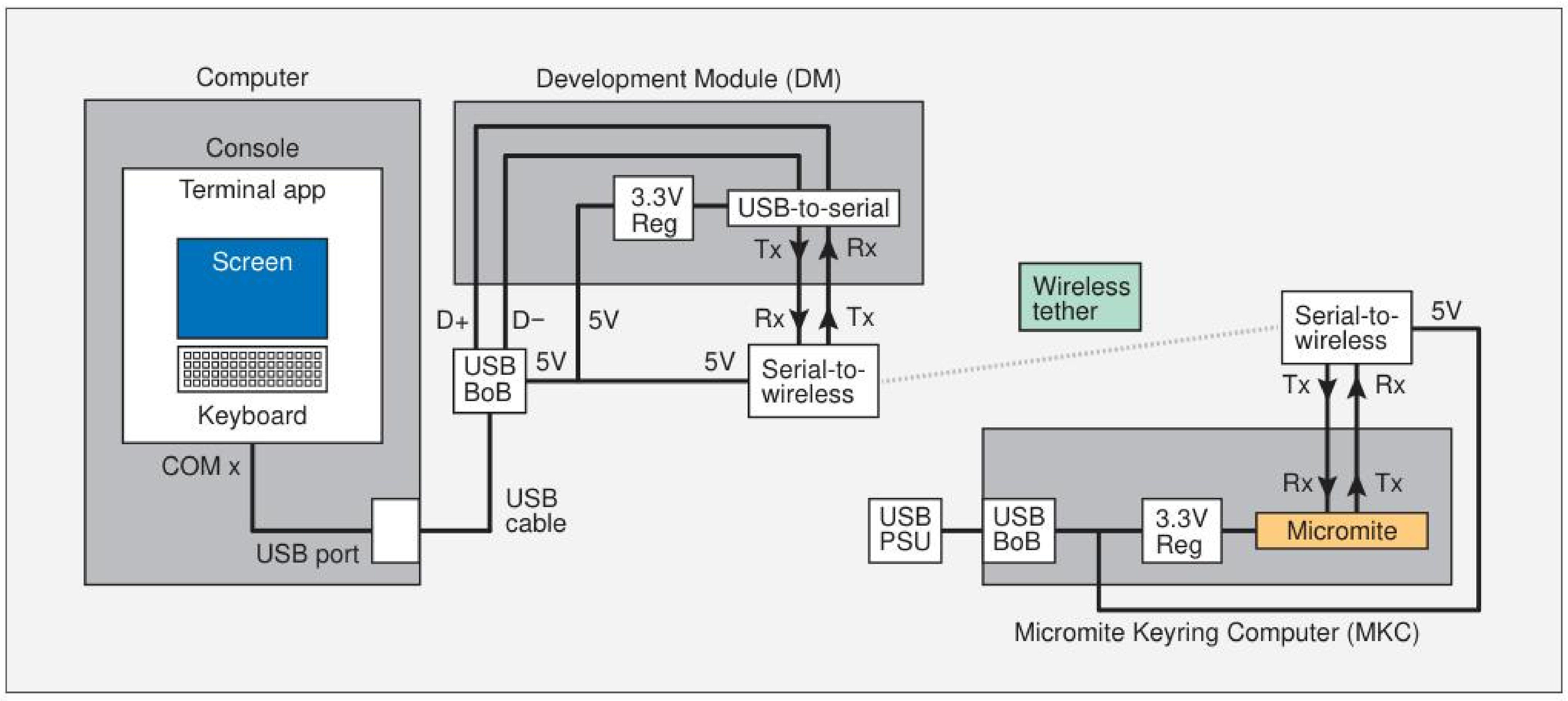


Fig.3. Adding a pair of USB-to-wireless modules to Fig.2 allows for an untethered wireless link between the computer and the Micromite.

protocol, and provides a much simpler serial protocol that has two physical connections: Transmit (Tx) and Receive (Rx). These are the connections that the Micromite serial console uses.

Now, without changing any connections, let's reposition the elements shown in Fig.1 to create Fig.2. This version highlights that we're using a 'serial-link tether' as opposed to a 'USB cable tether'. The important point here is that a wired serial link is much easier to convert into a wireless link compared to converting a USB-tether link. We do this by introducing a pair of serial-to-wireless modules, one at either end of the serial-link – see Fig.3.

Note that in this new layout, power and connectivity to the *DM* require the addition of a USB break-out-board (BoB) at the computer end. This is because the *DM* is no longer plugged into the *MKC*, which is where it got these connections previously. So, referring to Fig.2 and Fig.3, you can see that by adding a USB BoB at the computer end, the required power and data signals (D+ and D–) can be routed directly to the *DM*. At the Micromite end, we just need to power the *MKC* from a 5V USB power source.

That's the theory; now a quick look at a suitable serial-to-wireless module.

HC-05 Bluetooth module

The HC-05 is a low-cost Bluetooth serial-to-wireless module with all the functionality we need (see Fig.4). For now, simply consider a single HC-05 module as having two interfaces: a serial interface (Tx-pin and Rx-pin), and also a wireless Bluetooth interface. In use, any serial data sent (from an external device) to the HC-05's receive pin (Rx) is wirelessly transmitted via Bluetooth (ideally to a paired Bluetooth device). Likewise, any Bluetooth data received by the HC-05 (from a paired Bluetooth device) is sent out serially on the transmit pin (Tx). Note that all of this happens without us having to worry about any of the technicalities of the Bluetooth protocol.

Now consider having two HC-05s paired to each other via Bluetooth – see Fig.5. What the above translates into is that any serial data sent to the Rx pin on the left-hand HC05 is then automatically outputted on the Tx pin of the right-hand HC-05. Likewise, serial data received on the Rx pin on the right-hand HC-05 is automatically outputted from the Tx pin on the left-hand HC-05. In effect, we have created an untethered-wireless serial-link; and this is exactly what we need to replace the hard-wired 'serial-link tether' (see Fig.2 and Fig.3).

Some of you with Bluetooth built into your computer may be wondering why we're not using it at the 'computer end' of the wireless link, and therefore just use a single HC-05 at the 'Micromite end'. You can indeed do this, but in practice there are many potential complications, such as: incompatible Bluetooth versions, driver issues, drop-out problems, other Bluetooth device(s) already paired to your computer's Bluetooth… and so on. In the design presented here we will eliminate all of these potential issues by simply using a pair of low-cost HC-05 modules that form a dedicated wireless link to your Micromite.

HC-05 hardware overview

Referring to the diagram in Fig.4a, the hardware elements of the HC-05 that we need to concern ourselves with are the:

- Six pins on a 0.1-inch single-row header (discussed below)
- Small red status LED (flashes at different speeds to indicate various things – useful during testing)
- Miniature button (to put the unit into configuration mode).

Note that not all HC-05 modules come with the miniature button soldered into place (but the two solder pads will always be there – Fig.4c). If it is absent, use the two solder-pads to connect a 2-way 0.1-inch pin header, and use a jumper to mimic the button being pressed (required when we come to configure the HC-05 modules). Turning to the six pins on the module:

- **EN** is the enable input. If left unconnected (as we are doing), or if tied high (+V) it will allow the HC-05 to operate. If taken low (linked to GND), the unit is disabled (and enters low-power mode)
- **GND** and **VCC** supply the required power (3.6V to 6V)
- **TXD** and **RXD** are the two serial-interface pins
- **STATE** is an output (active high) that indicates successful connection status of the Bluetooth module (a totally different function to the small on-board red LED).

Notes regarding Bluetooth and the HC-05

If you have used Bluetooth before you will no doubt recognise terms such as: 'master', 'slave', 'paired', 'bind', and 'passkey'. Several of these terms will become relevant when we initially create our Bluetooth link

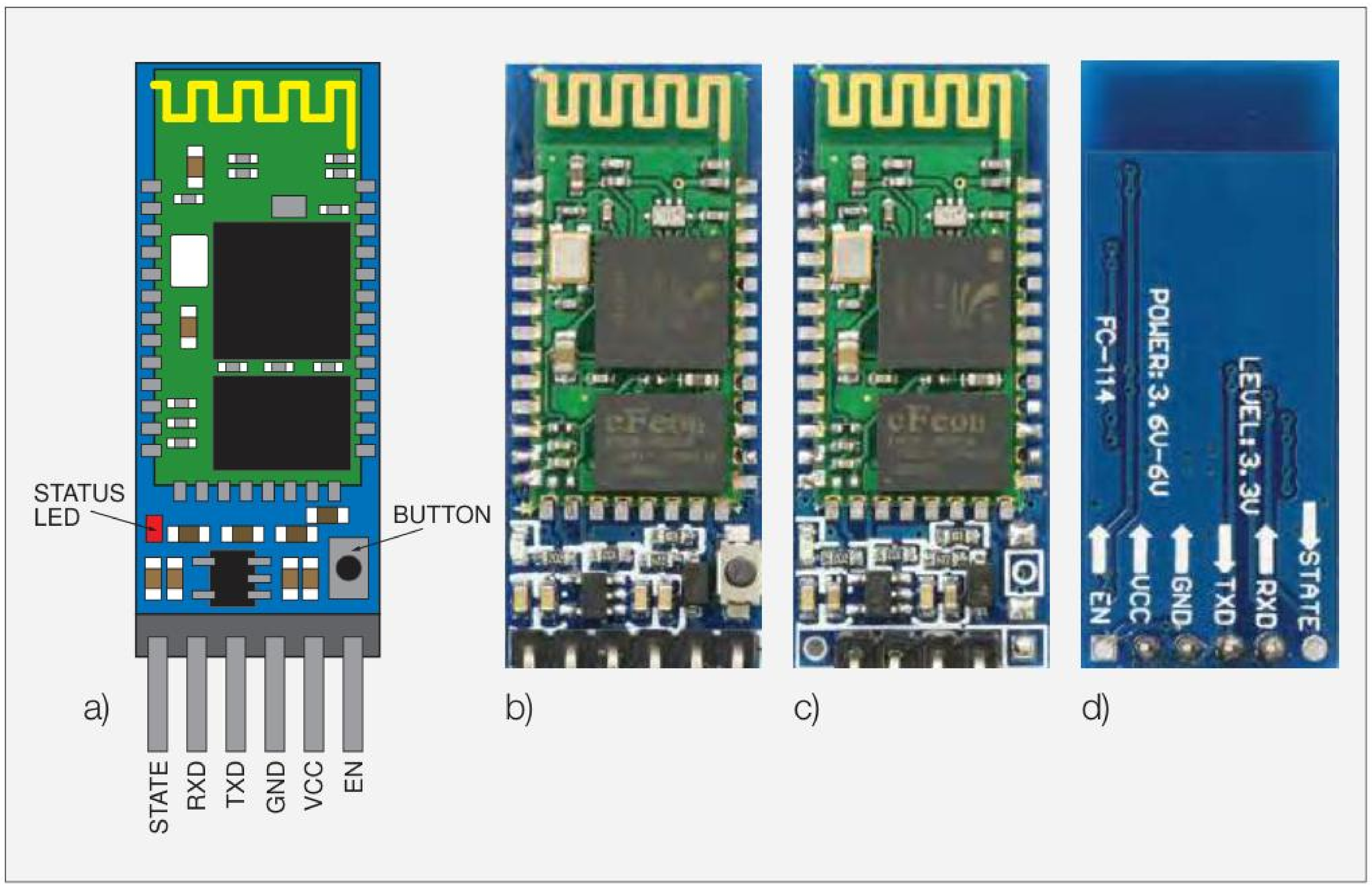


Fig.4. (a) Pinout of the HC-05 serial-to-Bluetooth module. Some have the button soldered in place (b), and some don't (c). Pins are labelled on the reverse of the module (d).

because they are part of the configuration that we will need to perform. Before we discuss the configuration in detail, there are several points that we need to highlight regarding Bluetooth and the HC-05 module:

- Bluetooth is a one-to-one wireless link that connects two devices
- To form a Bluetooth link, one end must be the 'master' Bluetooth device, and the other must be the 'slave' Bluetooth device. This means that two masters will not connect to each other, nor will two slaves
- For a master and a slave to connect to each other, they need to have the same 'passkey' (sometimes referred to as 'passphrase', or 'password')
- When two devices are communicating with each other via Bluetooth, they are regarded as being 'connected' and 'paired'. No other device can join in the communication (remember, Bluetooth is a one-to-one connection)
- The HC-05 has two modes of operation: 'configuration' (also known as AT mode), and 'communication' (also known as Data mode)
- In configuration mode, there are several settings that can be altered. These settings ultimately affect how the HC-05 behaves when in Data mode
- To enter configuration mode, the HC-05 module is powered up while pressing the module's button (or shorting the link on modules without a button)
- Configuration settings are sent to the HC-05 via the serial link (Rx and Tx pins) at a fixed baud rate of 38,400
- Configuration settings can be sent to the HC-05 from any terminal application, such as TeraTerm (use the same terminal app you currently use when connecting to the Micromite via the *DM*)
- To configure a setting, use the format: `AT+setting_name=value` (this is why configuration mode is also referred to as 'AT mode')
- To interrogate a setting's current value, use the format: `AT+setting_name?`
- All settings on the HC-05 are stored in Flash memory. This means that they will be remembered even when the HC-05 is powered off
- Settings only need to be set up once; when all settings have been configured correctly, the HC-05 only needs to be used in Data (communication) mode
- However, any setting(s) can be changed at any time by re-entering configuration mode
- Two correctly configured HC-05 modules can be made to automatically pair with each other (by 'binding' the master to the address of the slave).

If any of the above is unclear, please don't worry; we will cover most of these points as we work through the assembly and configuration of the Bluetooth link. After you have finished reading this article, do reread the above list.

Tasks to implement the Bluetooth link

In order to create the wireless Bluetooth serial-link in Fig.5, we need to configure one HC-05 as a master, and the other HC-05 as a slave. Then we need to ensure that we bind them together so that they automatically connect to each other. Note that once they are both configured, it does not matter which HC-05 module in Fig.5 is the 'master', and which is the 'slave'. This is because the two modules together form a two-way (duplex) serial-link (Tx>Rx and Rx<Tx). Do not fall into the misconception that one module is a 'transmitter', and the other is a 'receiver' – each is both.

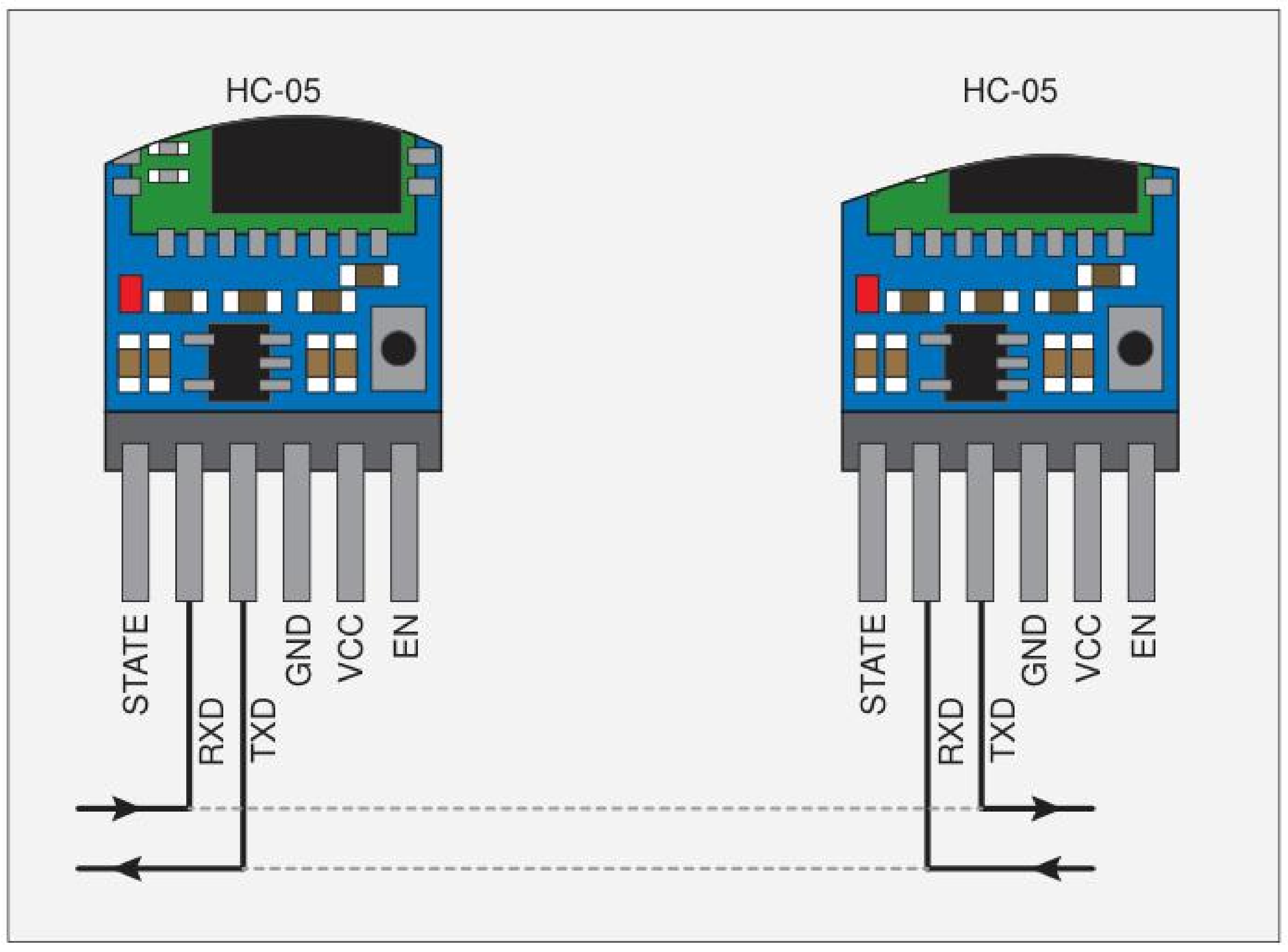


Fig.5. Using two HC-05 modules to create a wireless Bluetooth serial-link; effectively replacing a wired serial link (represented by the dotted lines).

To be able to configure the two HC-05 modules in the first place, and then go on to physically use them with our *DM* and *MKC* (as shown in Fig.3), we need to work through the following sequence of tasks, T1 to T9:

T1 Build an HC-05 Bluetooth interface (with BoB) for the *DM* (computer end)
T2 Build an HC-05 Bluetooth interface for the *MKC* (Micromite end)
T3 If either HC-05 is without the miniature button, then add a jumper link
T4 Set terminal application settings to HC-05 requirements
T5 Configure one HC-05 as a slave
T6 Configure the other HC-05 as a master
T7 Set terminal application settings back to Micromite requirements
T8 Plug everything together
T9 Test the wireless Bluetooth link

Keeping the hardware simple

Looking through the list above, you will see that only the first three tasks involve any hardware construction – the rest is configuration and testing. Referring back to Fig.3, you can see that the serial-to-wireless module (HC-05) at the computer end connects directly to the USB-to-serial module (in the *DM*). Likewise, the HC-05 at the Micromite end connects directly to the *MKC*. So to keep the theme of 'simple plug-in modules' for the *MKC* and *DM*, and also to keep construction to the bare minimum, we only need to assemble two straightforward adaptor boards: one for the computer end that plugs into the *DM* (T1); and another for the Micromite end that plugs into the *MKC* (T2). Both adaptors will require a 6-way socket to allow the HC-05 module to be plugged in. In addition, we will use an LED on each adaptor to show when the two Bluetooth modules are connected to each other (this will prove very useful when the modules are positioned in different locations). Everything else on each adaptor is just the required pins and/or sockets, to make the relevant connections. Fig.6 shows the assembled adaptors.

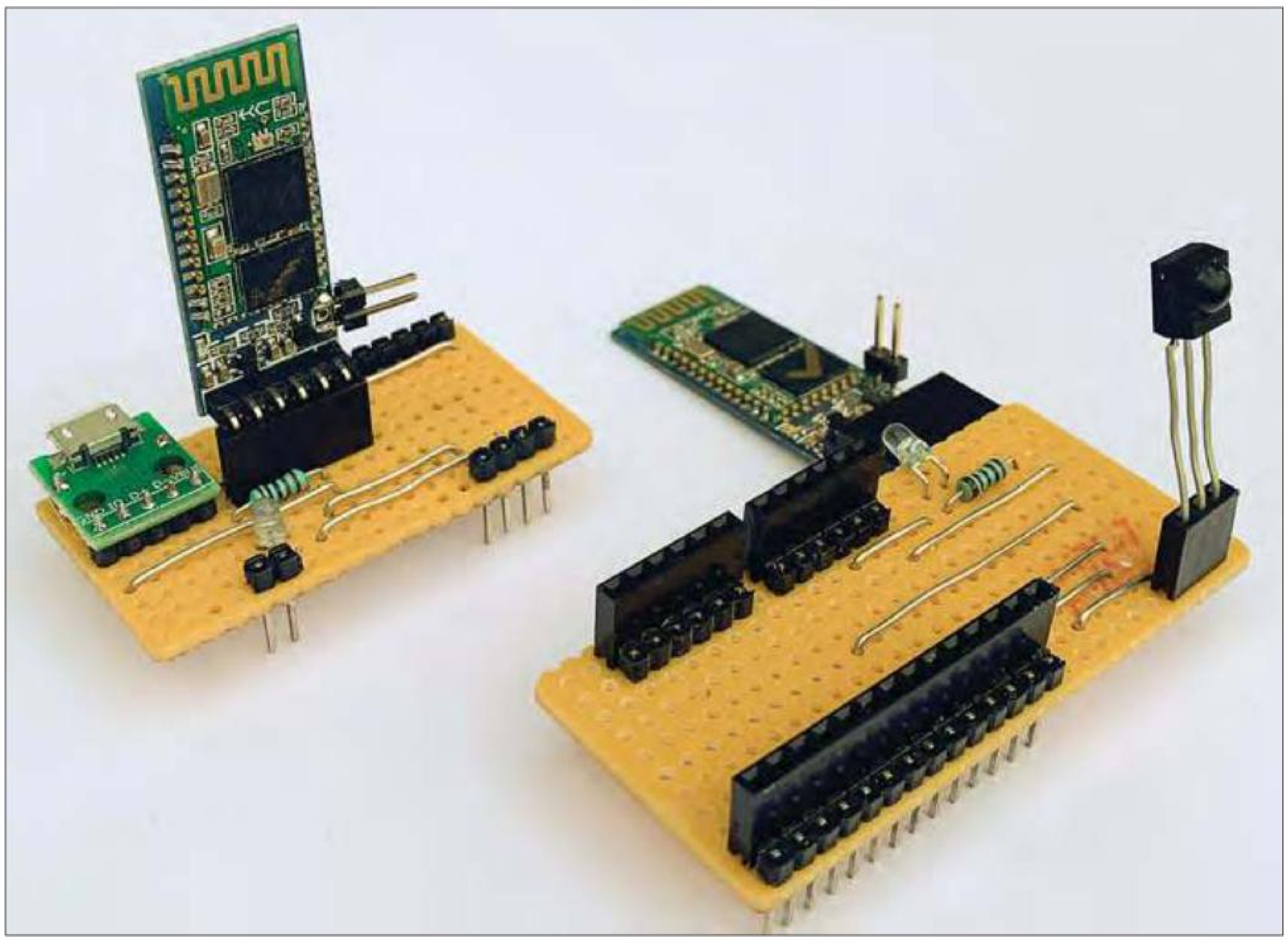
A really useful bonus (or rather, part of the design) is that the adaptor board at the computer end will also double up as the HC-05 configuration unit. Let's begin with the first assembly task.

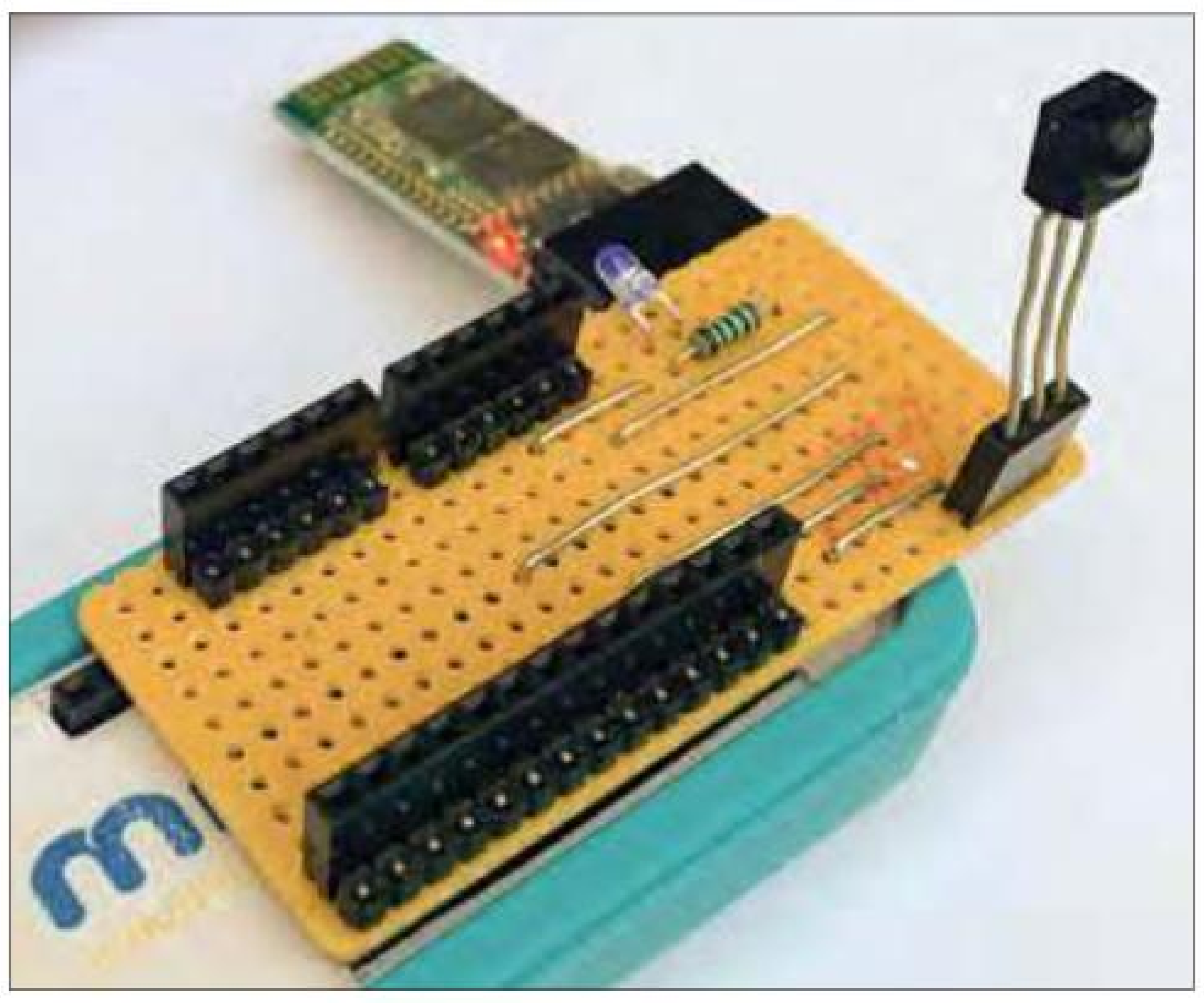
T1: HC-05 adaptor for the DM

This adaptor is designed to be plugged into the *DM* (from above) and needs the following specific requirements:

- Correctly positioned pins (downward facing) allowing the adaptor to be plugged into the *DM*.
- A socket for the HC-05 module. This is just a 6-way socket and allows easy insertion, and removal, of an HC-05 (especially useful during the configuration process).
- An LED and current-limiting resistor, which indicates that the HC-05 has successfully connected (via Bluetooth) to the other HC-05 module at the Micromite end. Since we are creating a Bluetooth link, we will use a blue LED (you can use any colour, but do use the appropriate resistor value)
- A USB socket – this allows connection to the computer and is required because the *DM* does not have its own USB socket. We will use a USB BoB identical to the one used in the *MKC* (see Fig.3).

You can see from the schematic (Fig.7) that there is nothing complex – it's just a matter of making the required connections between the USB BoB, the HC-05, the LED, and the *DM* (as outlined in Fig.3).





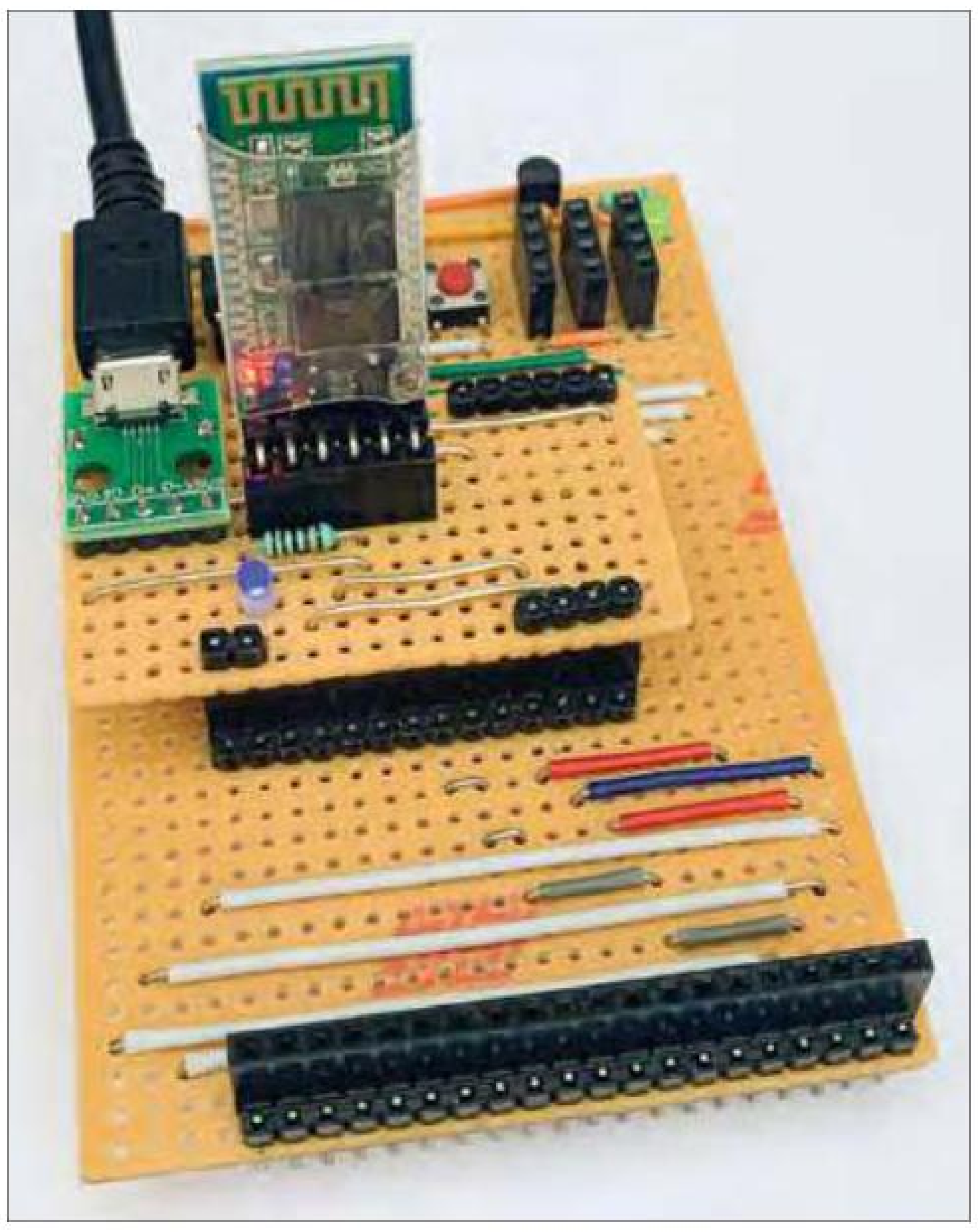


Fig.6. (left) The completed adaptors: note here that the buttons are not in place on the HC-05 modules and hence one horizontal, and one vertical 2-way pin header has been added onto each HC-05. (right) The adaptors in use: the one with the USB BoB plugs into the *DM*, and the other adaptor with the 3-way IR socket plugs into the *MKC*.

The USB socket (on the USB BoB) has five pins broken out onto a 0.1-inch header – although only four are used. The outer two pins supply 5V power to the HC-05 module and the *DM*. Two other pins from the BoB (D+ and D–) are routed directly to the *DM* (to the USB-to-serial module). The Rx/Tx from the USB-to-serial module (*DM*) are routed directly to the HC-05's serial pins (RXD and TXD). Finally, the active-high output from the HC-05's STATE pin directly feeds the blue (connection status) LED via a 1kΩ current-limiting resistor.

Fig.8 shows the stripboard, along with the relevant track-cuts, wire-links, and component placements. We will not go into step-by-step detail – it's a straightforward build. Simply cut the correct size board (19 tracks by 10 holes); make the seven track-cuts; insert the six wire-links, 1kΩ resistor, LED (long lead (anode) into hole H7) and mount the four banks of downward-facing pins (that will plug into the *DM*). Next, mount the 5-way pinstrip (for the BoB), and the 6-way socket (for the HC-05). Finally, solder the BoB onto the 5-way upward-facing pinstrip – be careful to get the correct orientation (the USB socket is upper-most, meaning the GND pin should be closest to the left edge of the stripboard). It is also worth soldering two pieces of wire to strengthen the USB BoB onto the stripboard (positions B1 and B5).

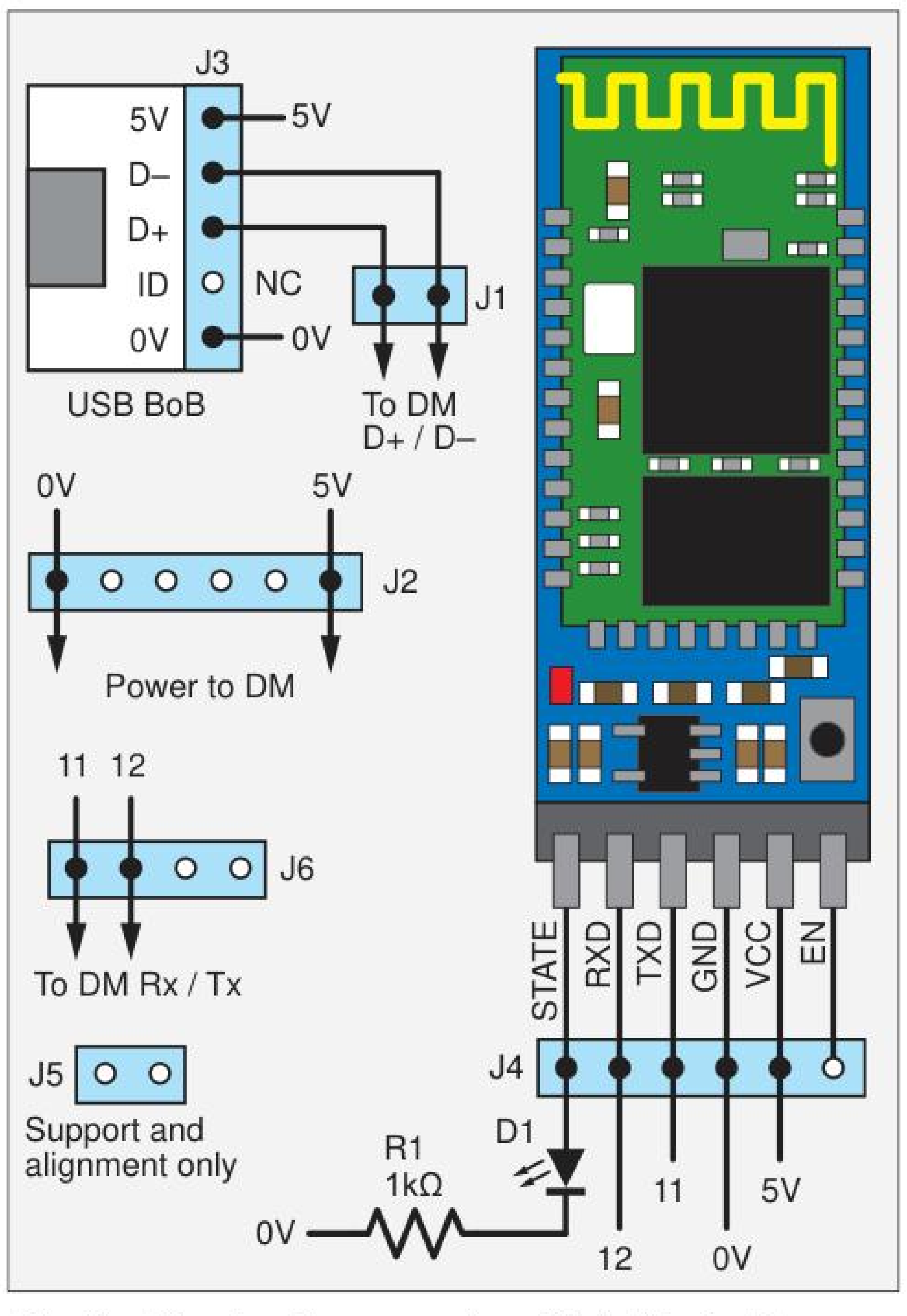


Fig.7. Block diagram for *DM*. Note the additional USB BoB that is required.

And that's all there is to it. Throughout construction, do the usual checks to ensure that track-cuts are across the full width of the track, and that there are no shorts between adjacent tracks. Correct anything that doesn't look right.

To test the adaptor, carefully insert it into your *DM* with the orientation shown in Fig.6, but do not install an HC-05 yet. Connect it via your micro-USB lead to your computer, and launch your terminal app exactly as if you were using the *MKC*. Press some keys and you should see the *DM's* red LED (data) flash as you press them. However, nothing should appear on the screen. If you do *not* see the *DM's* red LED flash then stop and check everything. There is not much that can cause an issue, so start by ensuring that the *DM's* green LED (power) is lit; ensure that the adaptor is inserted into the *DM* correctly; check the soldering on the adaptor, as well as the track-cuts and wire-links; and finally check that the terminal app is connecting to your usual COM port (as if you were connecting to your *MKC*).

Once you can see the red LED flash each time you press a key, then you can continue. Grab a short piece of wire (or a component leg off-cut) and temporarily insert it between the second and third holes on the 6-way HC-05 socket (link E8 to E9). This will connect the Tx output from the USB-to-serial module (on the *DM*) back to the Rx input. So now when you press keys you should see the characters appear (echo) in your terminal app. Note that the Enter key does not move down

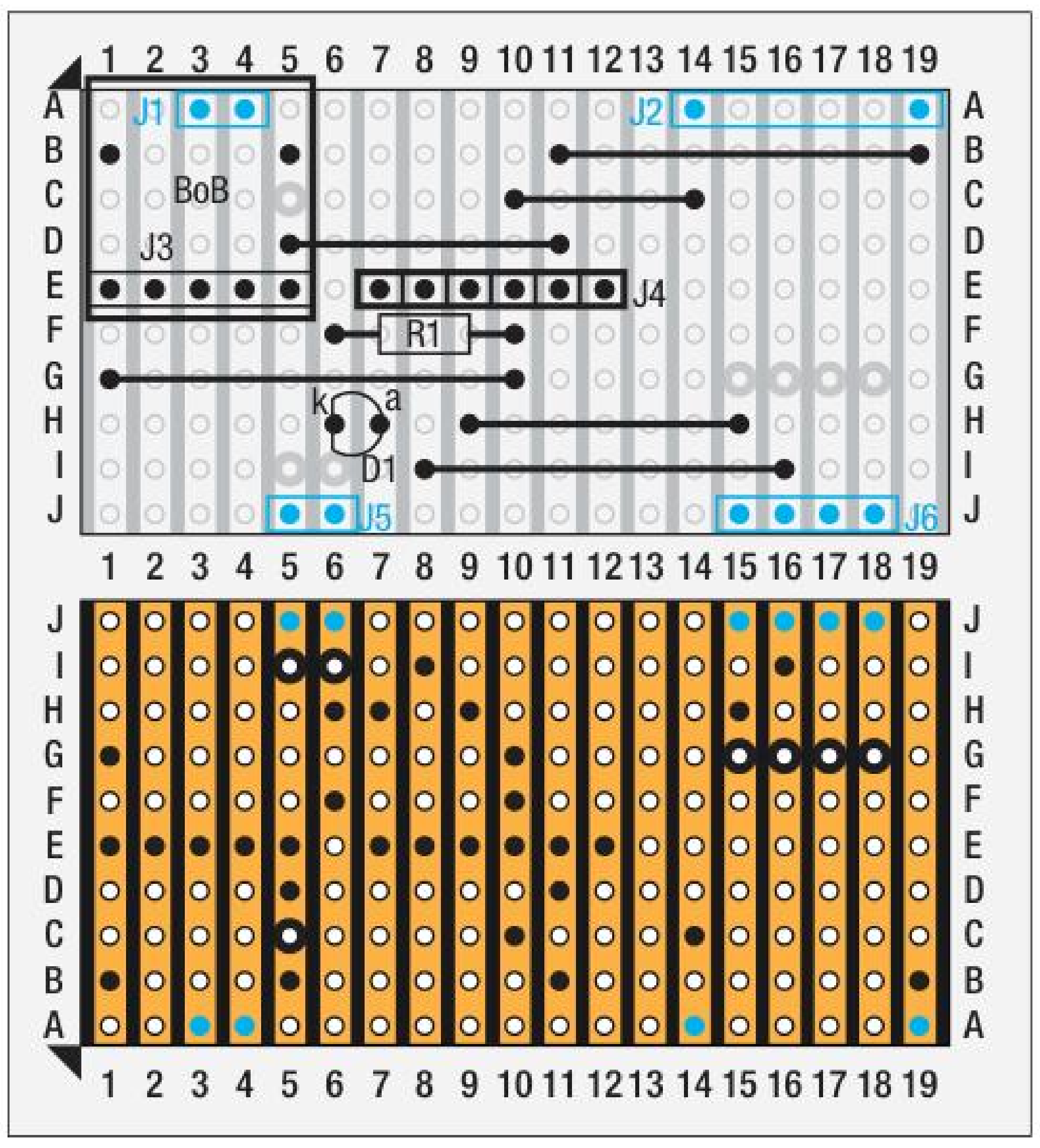


Fig.8. Stripboard for *DM*: (top) component side (bottom) conductor strip side. (Blue indicates downward facing connector pins on underside – see photos in Fig.6.)

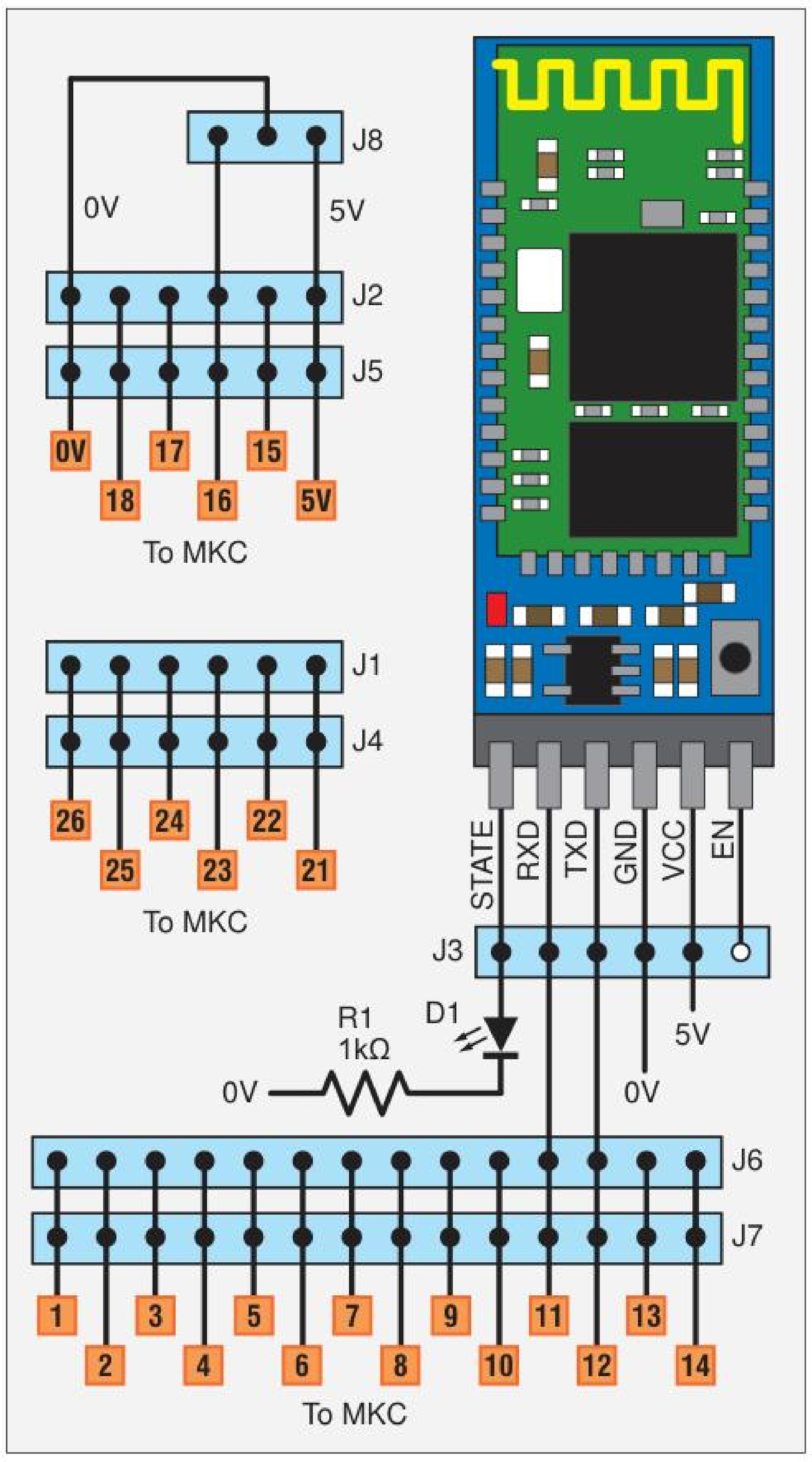


Fig.9. Block diagram for the *MKC* adaptor

a line (ie, it does not perform a line feed (LF)). Instead, it will just position the cursor at the beginning of the current line (ie, perform a carriage return (CR)). If you don't see the characters appear when you press keys then perform the usual checks.

Next, remove the temporary link from the second and third position and insert it between the first and fifth holes on the 6-way socket. This should result in the adaptor's LED being lit. If it isn't lit then check the LED circuit, and if you have access to a multimeter, carefully check that you have close to 5V present at the fifth position. Correct anything wrong, and ensure that you do see the LED on the adaptor lit (when positions 1 and 5 on the HC-05 socket are linked) before moving on.

As a final test, remove the temporary test link and carefully insert an HC-05 module (see Fig.6 for orientation). You should see the small red LED on the HC-05 flash.

That concludes testing for now. We will come back and use this adaptor in task T5; but for now, unplug everything from your computer, and let's build the other adaptor.

T2: HC-05 adaptor for the MKC

This adaptor is designed to be plugged into the *MKC* (from above). However, it must also allow other hardware modules to be plugged in (normally other modules would be plugged directly into the *MKC*). So this is a pass-through adaptor and needs to have the following requirements:

- Correctly positioned pins (downward facing) allowing the adaptor to be plugged into the *MKC*
- Upward facing sockets that mimic those on the *MKC* (allowing access to the Micromite's I/O pins, and other hardware modules to be plugged in)
- A socket for the HC-05 module. This is oriented horizontally so that when the HC-05 is inserted, it is not going to prevent other plug-in modules from being inserted (see Fig.6)
- An LED and current-limiting resistor for visual indication that the HC-05 has successfully connected to the HC-05 module at the computer end.

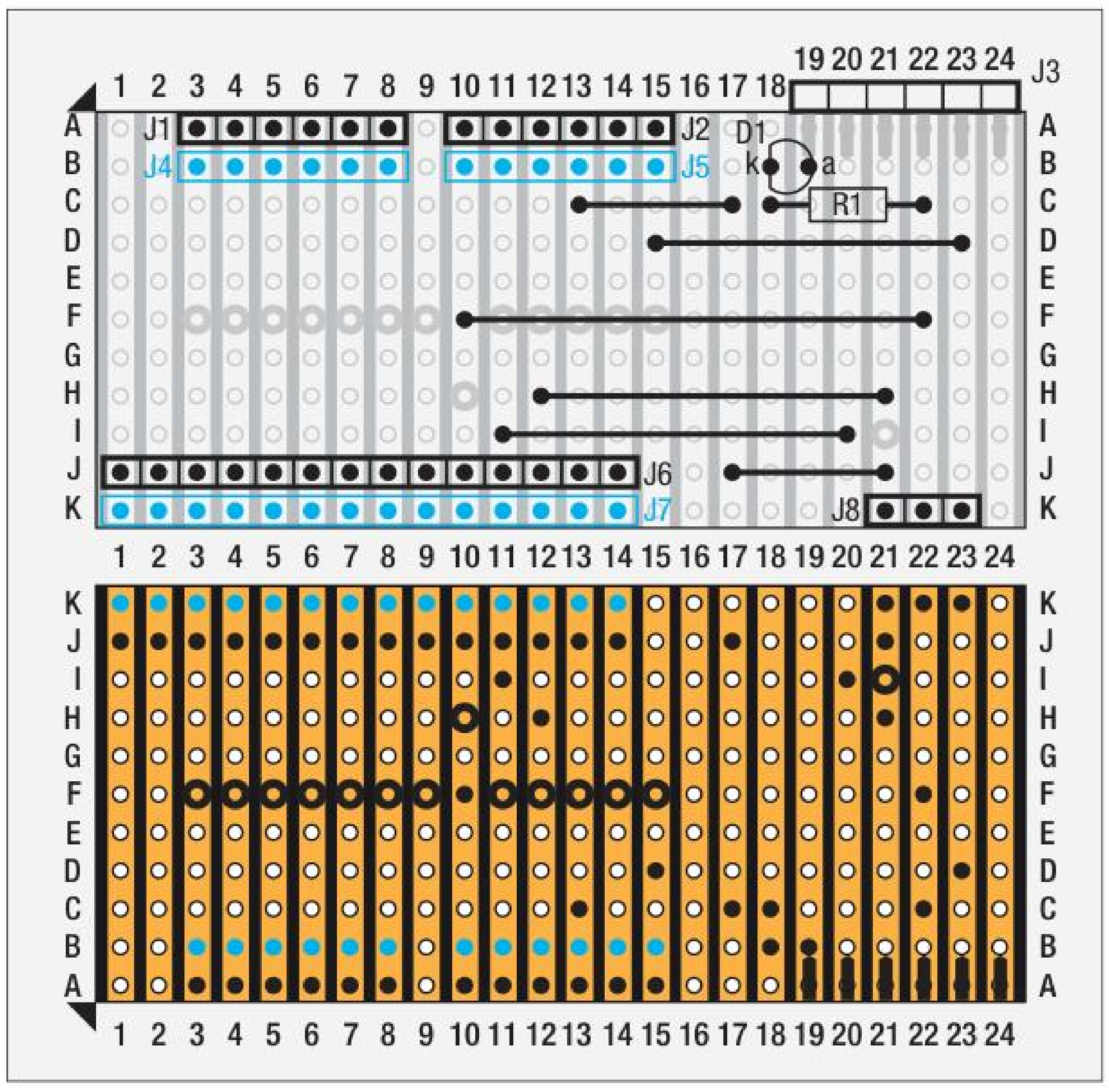


Fig.10. Stripboard for the *MKC*: (top) component side (bottom) conductor strip side. (Blue indicates downward facing connector pins on underside – see photos in Fig.6.) J3 soldered directly to copper strips.

The schematic for this adaptor is shown Fig.9. Once again, it is just a matter of making the required connections between the *MKC*, the HC-05 and the LED (as outlined in Fig.3). The 5V power for the HC-05 on this adaptor board is taken directly from the *MKC*. The HC-05's serial pins (RXD and TXD) feed directly to the *MKC's* Tx and Rx (console) pins. And, as before, the active-high output from the HC-05's STATE pin directly feeds the connection status LED via a 1kΩ current-limiting resistor. Note that a 3-way socket (J8) has been added that allows an infrared receiver to be inserted. This is not a requirement for the Bluetooth link, but it adds useful functionality if we want to use an IR receiver in a project.

The stripboard requirements are shown in Fig.10 – it uses a slightly bigger board (24 tracks by 11 holes). Make the 14 track-cuts; insert the six wire-links, 1kΩ resistor, LED (with the long lead (anode) in hole B19); mount the three banks of downward-facing pins (that will plug into the *MKC*); and mount the 6-way socket (for the HC-05). Note the HC-05 socket (J3) is soldered to the underside of the stripboard (not through holes), and with a horizontal orientation (see Fig.6). Finally, solder the four upward facing sockets. Perform the usual checks along the way to ensure the track cuts are correct, and no solder shorts have been introduced between adjacent tracks.

When you are happy that everything looks correct with this adaptor, carefully insert it into your *MKC* with the orientation shown in Fig.6. Again, do *not* install an HC-05. Then connect your *MKC* to your computer (or any 5V USB PSU) using your usual USB lead. Note that we are only testing for 5V on the adaptor, so there is no need to launch your terminal app for this test. If you have access to a multimeter you can carefully check that you have close to 5V present at the fifth position on the HC-05 socket (J3) (effectively point A23). Insert a short piece of wire into the first and fifth positions of the HC-05 socket to check that the adaptor's LED turns on. Perform the usual checks and rectify errors before moving on.

Now you may carefully insert an HC-05 module with the correct orientation (see Fig.6), and check the small red LED on the HC-05 flashes. If it doesn't flash then you will need to ensure you have a 5V supply going to the GND and VCC pins on the HC-05 socket. If 5V is there but the LED does not flash then it may be a faulty HC-05. (I mention this because I have had one – out of approximately 30 purchased – that turned out to be faulty.)

That concludes testing this adaptor; we will come back and use it in task T8 when we set up the complete Bluetooth link.

T3: No Button? Simply add a link

By default, when you power up the HC-05 it will be in communication (Data) mode; ie, it acts as a serial-to-wireless module – albeit one that may not actually be paired to another Bluetooth device. However, as mentioned above, to enter configuration mode you need to hold down the miniature button while powering up the HC-05. If both of your HC-05 modules have the button in place, then you can jump this section and move straight on to task T4. However, if either HC-05 has no button (and you do not have any spare miniature buttons to solder into position) then solder a two-way pin-header across the button pads. You can mount this header either horizontally, or vertically (see photos in Fig.6). Then, whenever you need to put the HC-05 into configuration mode, simply place a jumper link onto the header (to short out the button pads) before you power up the HC-05.

Now that all the hardware has been assembled, and some basic tests have been performed, we need to prepare your terminal app to configure the two HC-05 modules.

T4: Preparing the terminal app (for HC-05)

Until now, whenever you used your *MKC* for development, you first inserted the *DM* into it, and then connected this duo to your computer, from which you launched your terminal app. Everything (should have) then worked, and you saw either the usual Micromite welcome message, or cursor, or your running program. This all happens because you initially went through the process of making the necessary settings in your terminal app. These settings were then saved, allowing you to do a 'single click' (this was discussed in *Part 4*, May 2019).

In essence, the terminal app only talks to the USB-to-serial module (on the COM port assigned by the operating system). Nothing happens in the terminal app until something intelligent (ie, your *MKC*) is connected to the Tx and Rx pins on the USB-to-serial module. Whenever something is connected to these pins, successful communication can only take place if the settings in the terminal app match those required by the connected device (remember we had to set the speed to match the Micromite's `OPTION BAUDRATE` setting). So currently, your terminal app is specifically set up for communicating with your Micromite.

However, we have now replaced the *MKC* with an HC-05. This means we need to adjust some terminal app settings to allow successful communication with the HC-05 bcause its parameters are different to those used by the Micromite.

In the following discussion, we will be referring to TeraTerm as our terminal app. If you use a different terminal app then don't worry – you will use the same settings, but they will be found in different places within the menus. (Useful tip: before making any changes to the terminal app settings, take a picture of the screen showing the current settings – we will need to set them back to what they currently are once we have configured the two HC-05 modules. A camera on a smartphone can help here. Otherwise, simply write them down or take (and save) a screen grab.)

Before we adjust the terminal app settings, let's first connect some hardware for it to communicate with. Take the smaller adaptor and insert it carefully into your *DM*. Now connect this up to your computer in exactly the same way as if you were plugging in the *MKC/DM* combination. Check that the green power LED on the *DM* is lit. If an HC-05 module was already inserted into the adaptor, carefully remove it. Now launch your usual terminal application (exactly as if you were connecting to your *MKC/DM*). If everything goes well, then you will see your terminal screen open up (and it will be connected to the USB-to-serial module in your *DM*).

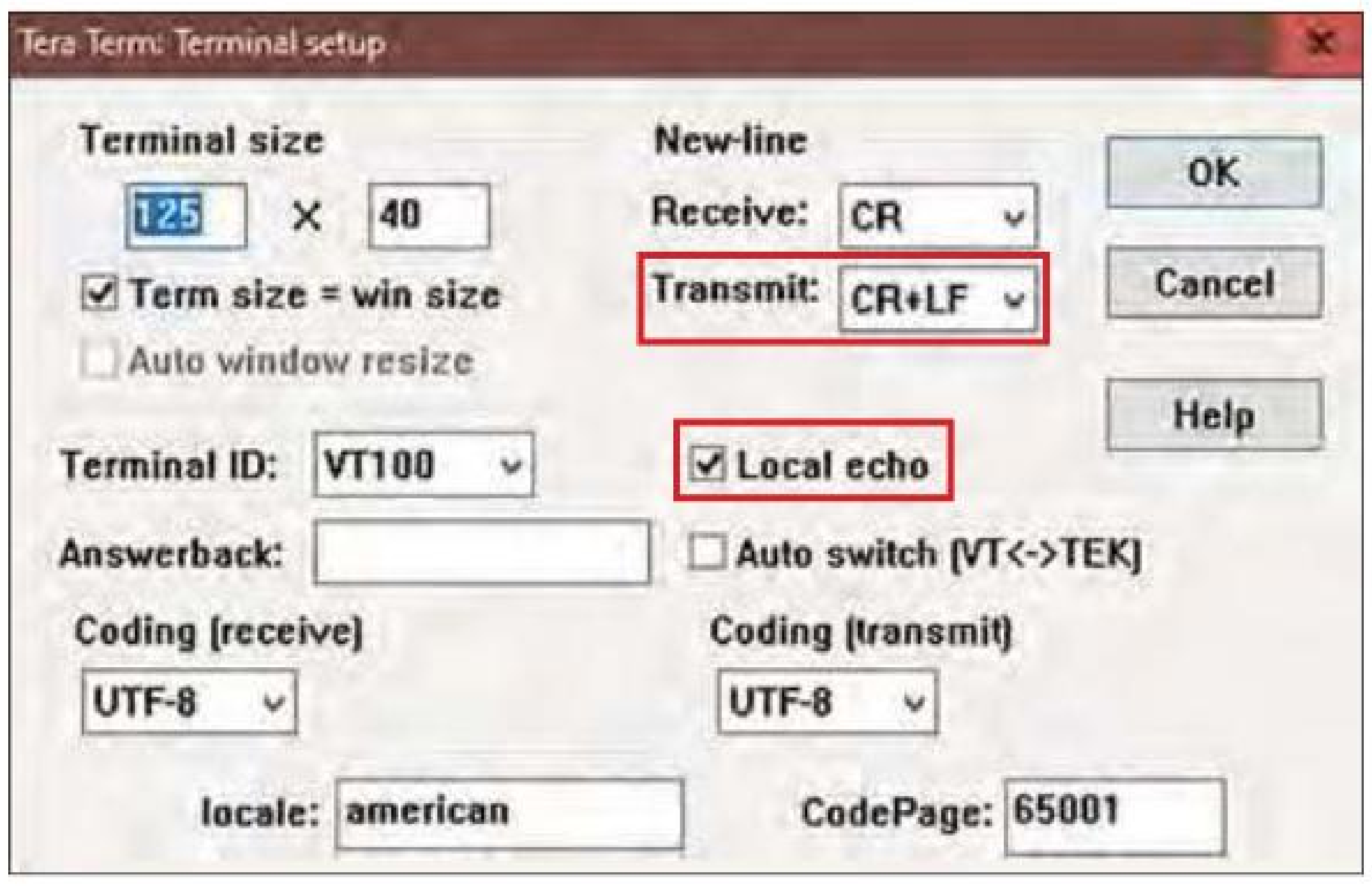
Do not expect to see anything on the terminal screen yet. The position we are trying to get to is: terminal app open and connected to your *DM*. If your terminal app doesn't connect for some reason, then check which virtual COM port your operating system has assigned (shutting down any open sessions of your terminal app first). Remember, at this stage simply treat everything as if you are connecting to the *MKC/DM* (all we have done is replace the *MKC* with a Bluetooth module).

Next, take one of the HC-05 modules and press the button (or add the jumper link) while you insert the HC-05 into the adaptor board. This should result in the small red LED on the HC-05 initially coming on, and then it should start flashing slowly. If this is the case, then the HC-05 is now in configuration mode. If the LED is flashing quickly, then either the button wasn't fully pressed, or the jumper link wasn't in place correctly. *Remember*, the button needs to be pressed while the HC-05 is inserted. If you're using a link, it can remain in place for now.

Once you have the small red LED flashing slowly, you are ready to move on and adjust the terminal app settings. First, we need to ensure the baud rate is set correctly. In TeraTerm, go to Setup/Serial port... and make a note of the current Speed setting – it is likely to be either 38,400, or 115,200. This is the speed that your terminal app communicates at with your Micromite. Now ensure that the Speed is set to 38,400, which the HC-05 uses (remember to record all current settings prior to changing them). Press OK to save.

Now go to Setup/Terminal and tick the Local echo option to set it. On this same screen, set the New-line, Transmit: option to CR+LF (see screenshot in Fig.11). Press OK to save. These are the only settings that need to be adjusted. Note that if you were to close your terminal app then these modified settings will be lost. So to save them more permanently, go to Setup/Save setup and then click the Save button. Henceforth, whenever you launch your terminal app, the settings will be pre-set for communicating with the HC-05.

Now for the crucial test – in the terminal window type `AT`, then press `Enter`; you should see an `OK` response. Note that it is case *insensitive*, so `at` is also valid. If you just see a blank line returned then simply repeat `AT` (`Enter`) until you see `OK`. Once you see this, it means that you are successfully communicating with the HC-05 module and you are now in a position to configure it (leave it plugged in for now). If you do not see an `OK` response then you will need to repeat the checks performed up to this point (ie, *DM* green power LED on, slow flash on HC-05 small red LED, 38400 baud, echo on and Transmit as CR+LF).



Terminal size 125 × 40
☑ Term size = win size
☐ Auto window resize
New-line
Receive: CR
Transmit: CR+LF
Terminal ID: VT100
☑ Local echo
Answerback:
☐ Auto switch (VT<->TEK)
Coding (receive) UTF-8
Coding (transmit) UTF-8
locale: american
CodePage: 65001

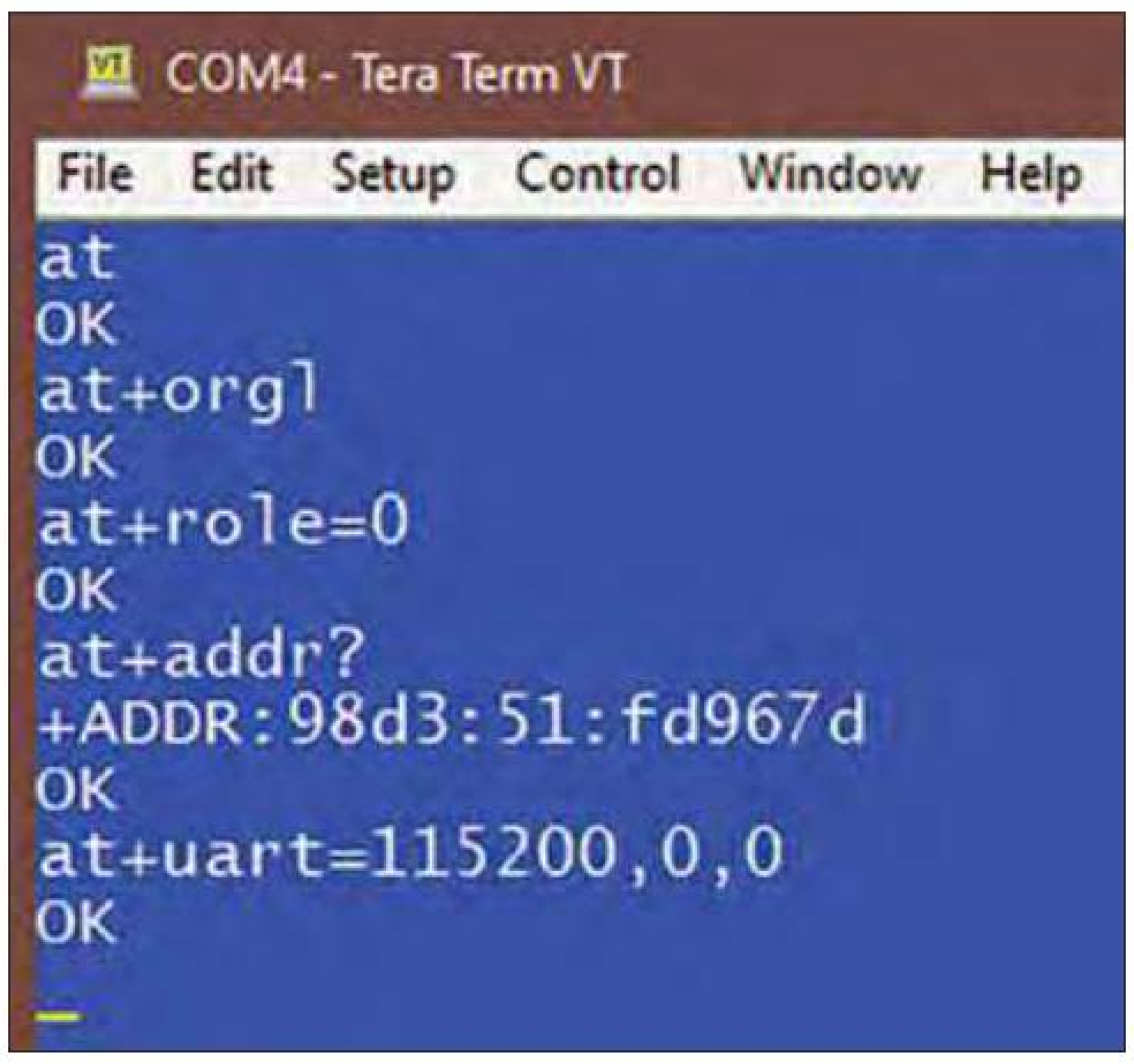
Fig.11. TeraTerm settings highlighted here need to be altered to enable successful communication with the HC-05. Also set TeraTerm's baud rate to 38400.

T5: Configure the slave HC-05

In this task, we will first reset the HC-05 to original factory defaults (just in case any settings have previously been changed). Then we will set the HC-05 to be a slave, followed by setting the baud rate of the HC-05 to the same value the Micromite uses (ie, what we noted down from the previous section). Finally, we will find the unique address of this HC-05 module so that the address can be programmed into the other HC-05 (the master). Doing this ensures the master only connects to this slave HC-05.

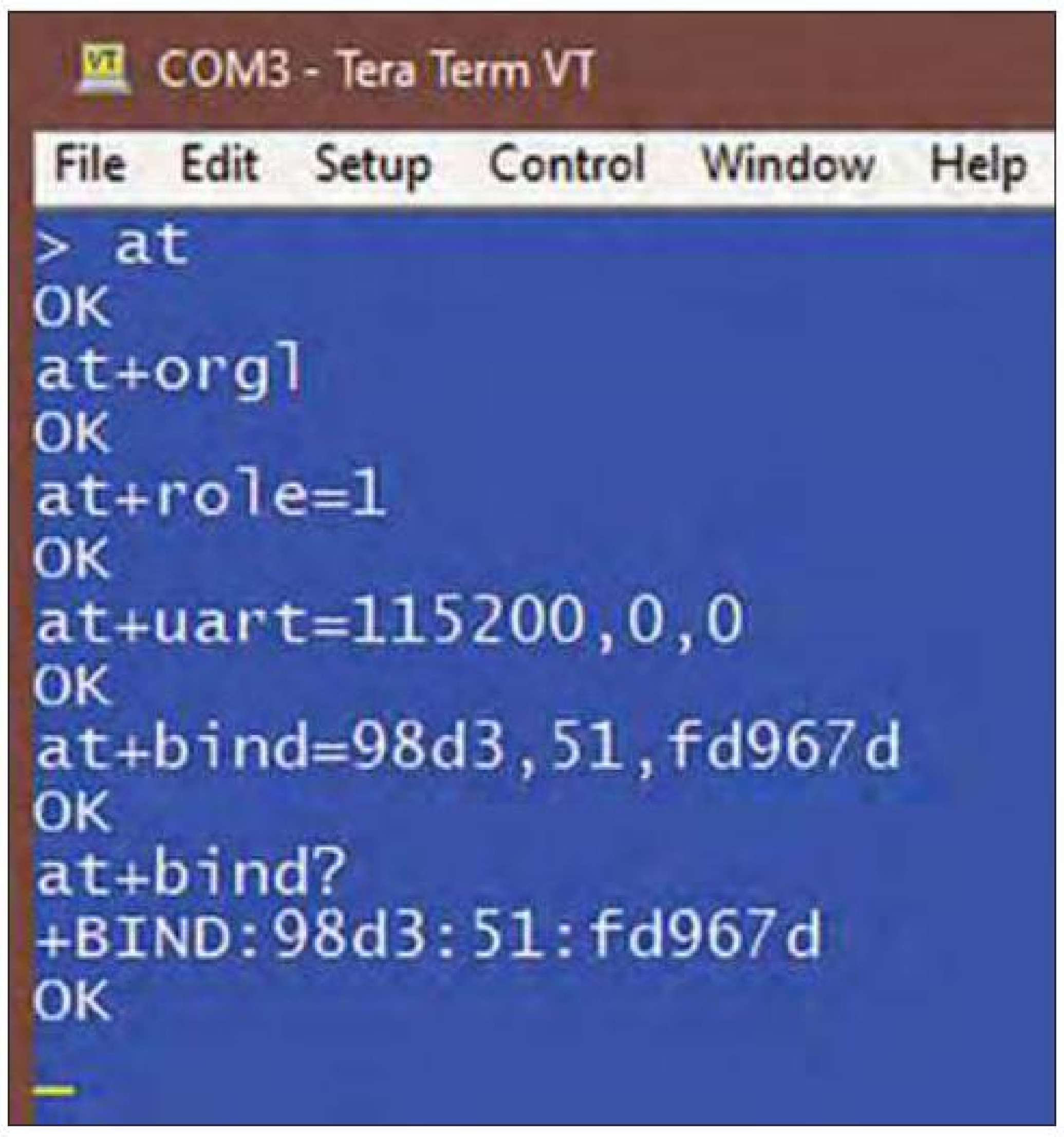
So, with an HC-05 inserted into the adaptor and in configuration mode (slow flashing red LED), let's now configure it to be the slave. You'll need to type in the commands ensuring you see an `OK` response at each stage (see Fig.12). Note that if you make a mistake when entering a command, using the `Delete` key, `BackSpace` key, or cursor keys will result in an error upon pressing the `Enter` key. If you do not see an `OK` response, or if you see any error message, then simply repeat the relevant command. So now type the following commands, each followed by `Enter`:

`AT+ORGL` Assigns original factory settings



```
at
OK
at+orgl
OK
at+role=0
OK
at+addr?
+ADDR:98d3:51:fd967d
OK
at+uart=115200,0,0
OK
_
```

Fig.12. The commands used to configure one HC-05 as the slave.



```
> at
OK
at+orgl
OK
at+role=1
OK
at+uart=115200,0,0
OK
at+bind=98d3,51,fd967d
OK
at+bind?
+BIND:98d3:51:fd967d
OK
_
```

Fig.13. The commands used to configure the other HC-05 as the master.

`AT+ROLE=0`	Defines slave mode
`AT+ADDR?`	Returns the unique address; write it down exactly as shown, for example: `98d3:51:fd967d`
`AT+UART=`**`115200`**`,0,0`	Sets the baud rate which must match your Micromite (ie, use the Speed value noted from the previous section); 115200 is just an example, you may be using a different speed.

Once all these steps have been completed, remove the HC-05 module and insert it into the *MKC* adaptor board. If you used a jumper link on the HC-05 to mimic the miniature button being pressed, then remove the link as the configuration for the slave HC-05 is now complete.

T6: Configure the 'Master' HC-05

In a similar fashion to what we have just done, plug the other HC-05 module into the *DM* (configuration) adaptor while pressing the button (or placing a jumper on the link). Once you see the slow flashing red LED to indicate it is in 'configuration' mode, type in the following commands, one at a time, followed by `Enter`, and ensuring you see an `OK` response to each command (see Fig.13):

`AT+ORGL`	
`AT+ROLE=1`	Defines master mode
`AT+UART=`**`115200`**`,0,0`	Use the exact same speed as you used for the slave
`AT+BIND=98d3,51,fd967d`	Use the slave address returned above – **NB, use commas to separate the address (not colons)**
`AT+BIND?`	Shows the address entered in the previous step. Check that it exactly matches the slave address returned in task T5 – this is an important step. The wrong address means the Bluetooth link will just not work. If it is different, then simply re-enter the above `BIND` command, but this time with the correct address.

This completes configuration of the master HC-05. If you used a link to enter configuration mode, then now is the time to remove it. Next, temporarily remove the HC-05 from the adaptor, then re-insert it. This takes the HC-05 module out of configuration mode, and puts it into communication mode.

T7: Set terminal application settings back to Micromite requirements

The last thing to do before testing the Bluetooth link is to reinstate the terminal app settings back to what they were originally (ie, what they were set to when used with your *MKC/DM*). In your terminal app, first set the correct baud rate (Setup/Serial port...) so that it matches what your Micromite uses. Then in Setup/Terminal, switch off echo (un-tick box) and also return Transmit: back to CR (and not CR+LF). Be sure to save the settings so that they are not lost when you shut down the terminal app. And that's it – we are now ready to test the Bluetooth link.

T8: Plugging everything together

If you have been following the tasks step-by-step, then you should currently have the *DM*, adaptor, and master HC-05 plugged into your computer (and powered up); and your terminal app successfully up and running. The HC-05's red LED should be flashing quickly indicating the Bluetooth link is waiting to connect to the Micromite end. The *MKC*, adaptor, and slave HC-05 should not yet be powered up. If this is not the case, then simply get to this position. Now apply 5V USB power to the *MKC* – you can use any 5V USB PSU for this, or even a mobile phone battery pack (you are aiming to set up all the elements as shown in Fig.3).

Ideally, position the *MKC* elements a short distance away from your computer (but no more than five metres for now). You should see signs of life as soon as you power up the *MKC* – this will be in the form of the slave HC-05 red LED initially flashing quickly. If there are no signs of life, then check that the *MKC's* blue power LED is lit. If not, then check the power supply. If it is lit, but there are no signs of life on the HC-05, then check that the HC-05 is inserted correctly, and also that the adaptor is inserted correctly into the *MKC*. If there is still an issue, then something has worked loose from when you tested the adaptor as part of task T2. In this case, perform the T2 checks once again.

T9: Wireless Bluetooth link test

If all is well, then in just a few seconds of powering up both ends, the two modules should automatically link together. Both adaptor LEDs being lit will indicate this. If this is the case then all is good! If both adaptor LEDs do not light up then you will need to go back and re-run all the checks in tasks T1 to T8.

Once both adaptor LEDs are lit you should find that you can now interact with your Micromite in the usual manner. Try typing something like `PRINT TIME$` (`Enter`) and check that all works as normal. If you don't see any response (but the adaptor LEDs are both lit), then is it because you have a piece of automatically running code? If so, then use `Ctrl-C` to return to the command prompt.

If all has worked out successfully, then congratulations – you have now 'set your Micromite free'. You are now able to use your Micromite as if it were plugged directly into your computer. If you do have any issues that you just can't resolve, then simply drop me an email explaining your situation and I will be happy to help you.

Upgrading your Mood Light to Bluetooth control

Here's a quick exercise to demonstrate remote interaction with the Micromite. Load the *Mood Light* software from last month into your *MKC* (`AUTOSAVE`, paste, `Ctrl-Z`). Then use four jumper wires to connect the Blinkt! module to the relevant *MKC* pins (via the sockets on the *MKC* adaptor) – refer to last month's article for the exact pins (SPI Clk, SPI Data, 0V and 5V). Then insert an IR receiver (TSOP) into the 3-way IR socket that we implemented on the *MKC* adaptor board; and finally type `RUN` (`Enter`).

You should now see the Blinkt! light up green. Press some buttons on the 44-button IR remote and you should see that the *Mood Light* works as normal. We are not adding any new features here – all we are doing is allowing the *Mood Light* to be located wherever you want, but with the power to edit the *Mood Light* code remotely from your computer. Try it – press `Ctrl-C`, `EDIT`, and check that you see the code. Now type in commands at the command prompt to change the colour of the *Mood Light* remotely (this is an exercise for you to work out...).

Now it's over to you to think of some other uses for your wireless Bluetooth link. Have Fun!

Next month

Next month, we will show you just how easy it is to add an inexpensive colour touchscreen to your *MKC*. We will then use it to simulate the classic board game *Mastermind* – all remotely controlled, of course!

Visual programming with XOD

By Julian Edgar

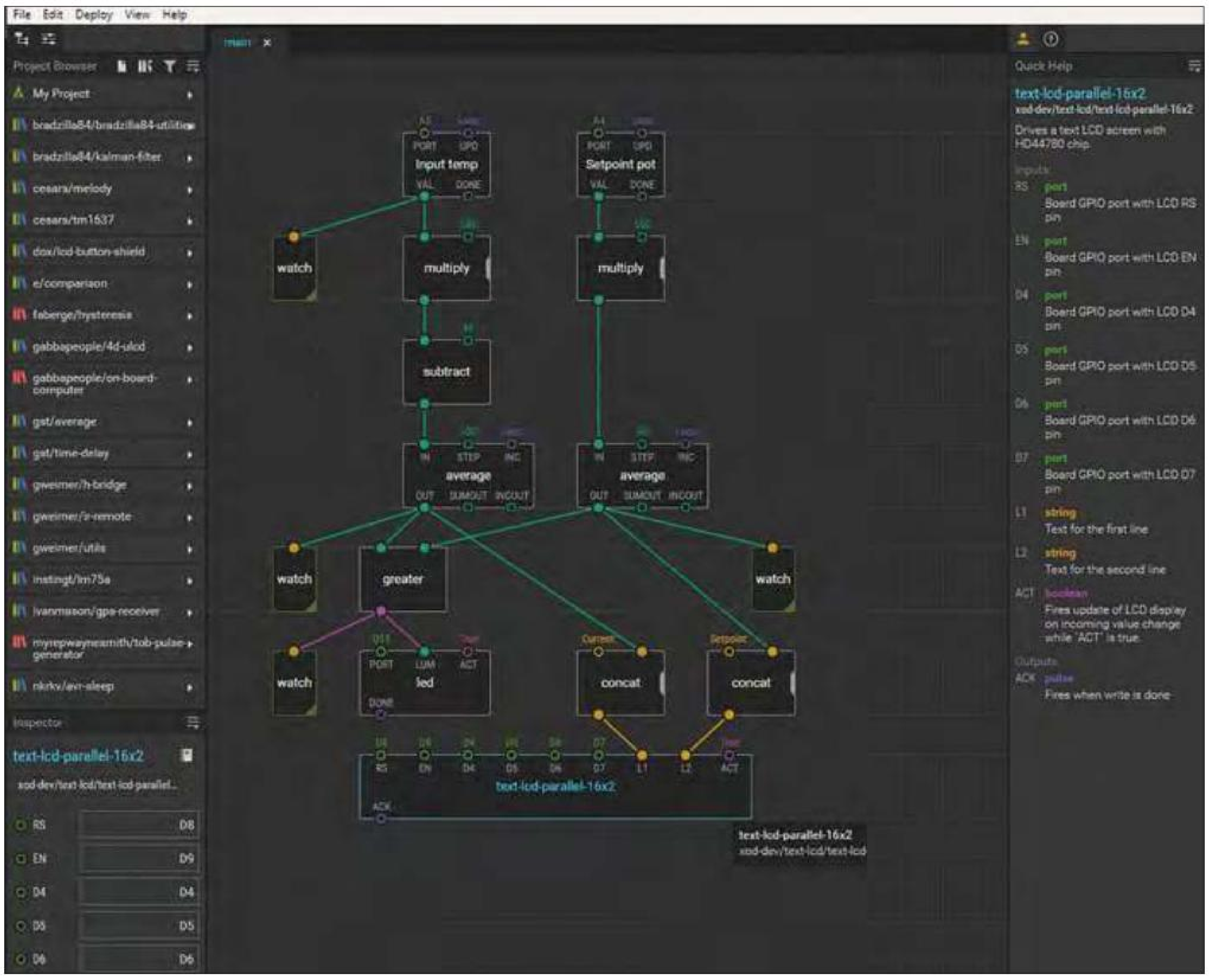
Introducing XOD visual programming for Arduino

I have a confession to make: the massive and growing use of programmable microcontrollers in hobby electronics has largely passed me by. Why? Well to be honest, I find listings of code terribly off-putting – one character in the wrong place and the program won't run; and anyway, who wants to learn lots of arcane instructions before you can get a microcontroller board to do even the simplest thing? (OK, lots of you! But others, like me, just want to get a quick digital fix!)

Well, in the last few months I've had an epiphany – and it's all due to XOD (pronounced 'zod'). XOD is a free visual programming software package for Arduino. There is no code – you simply place pre-built blocks on the screen, and then 'wire' them together. If you can understand a simple circuit diagram, you can program an Arduino in XOD.

I'm in no doubt that an expert code writer would find many deficiencies in XOD, but for someone without that background who wants to quickly get Arduinos doing real-world things, it's just wonderful. Furthermore, the level of support is good. Rather than hundreds of poor YouTube videos, the software has an inbuilt tutorial with 42 clear and relevant lessons. These take you from zero knowledge through to possessing enough knowledge to build quite complex control systems. There is also a user forum that I have found supportive and helpful.

Finally, XOD allows you to emulate the program on-screen – that is, you can see how the program will run, even as you write it. This makes it far easier to debug your program.


File Edit Deploy View Help
Project Browser
My Project
bradzilla84/bradzilla84-utilities
bradzilla84/kalman-filter
cesars/melody
cesars/tm1637
dox/lcd-button-shield
e/comparison
faberge/hysteresis
gabbapeople/4d-ulcd
gabbapeople/on-board-computer
gst/average
gst/time-delay
gweimer/h-bridge
gweimer/ir-remote
gweimer/utils
instingt/lm75a
ivanmason/gps-receiver
myrepwayneamith/tob-pulse-generator
nkrkv/avr-sleep
Inspector
text-lcd-parallel-16x2
RS D8
EN D9
D4 D4
D5 D5
D6 D6
Input temp
Setpoint pot
watch
multiply
subtract
average
greater
led
concat
text-lcd-parallel-16x2
Quick Help

text-lcd-parallel-16x2
xod-dev/text-lcd/text-lcd-parallel-16x2

Drives a text LCD screen with HD44780 chip.

Inputs
RS port Board GPIO port with LCD RS pin
EN port Board GPIO port with LCD EN pin
D4 port Board GPIO port with LCD D4 pin
D5 port Board GPIO port with LCD D5 pin
D6 port Board GPIO port with LCD D6 pin
D7 port Board GPIO port with LCD D7 pin
L1 string Text for the first line
L2 string Text for the second line
ACT boolean Fires update of LCD display on incoming value change while 'ACT' is true.
Outputs
ACK pulse Fires when write is done

The XOD programming screen. In the main work area is the program ('sketch') that's being built. Here the LCD programming box has been highlighted – the left-hand lower column on the screen shows the configurable settings for the LCD and the right-hand column displays the help file for the LCD.

Getting the software

The software can be downloaded from **https://xod.io/downloads/** and is available for Windows, macOS and Linux platforms. (If you don't want to initially download the desktop version, you can also explore the software a little by using a browser version.) Prior to downloading, or using the browser version of XOD, you will be required to create an account with your email address and a password. A reminder – XOD is completely free, so you don't get a version that is incomplete or hamstrung in what it can do.

Installing and running the software is straightforward; the only issue I had was specifying the correct Arduino board when uploading to the board for the first time. For an Arduino Uno or clone, select the default 'Arduino/Genuino Uno' option. You might also find that you need to specify the correct PC communication port.

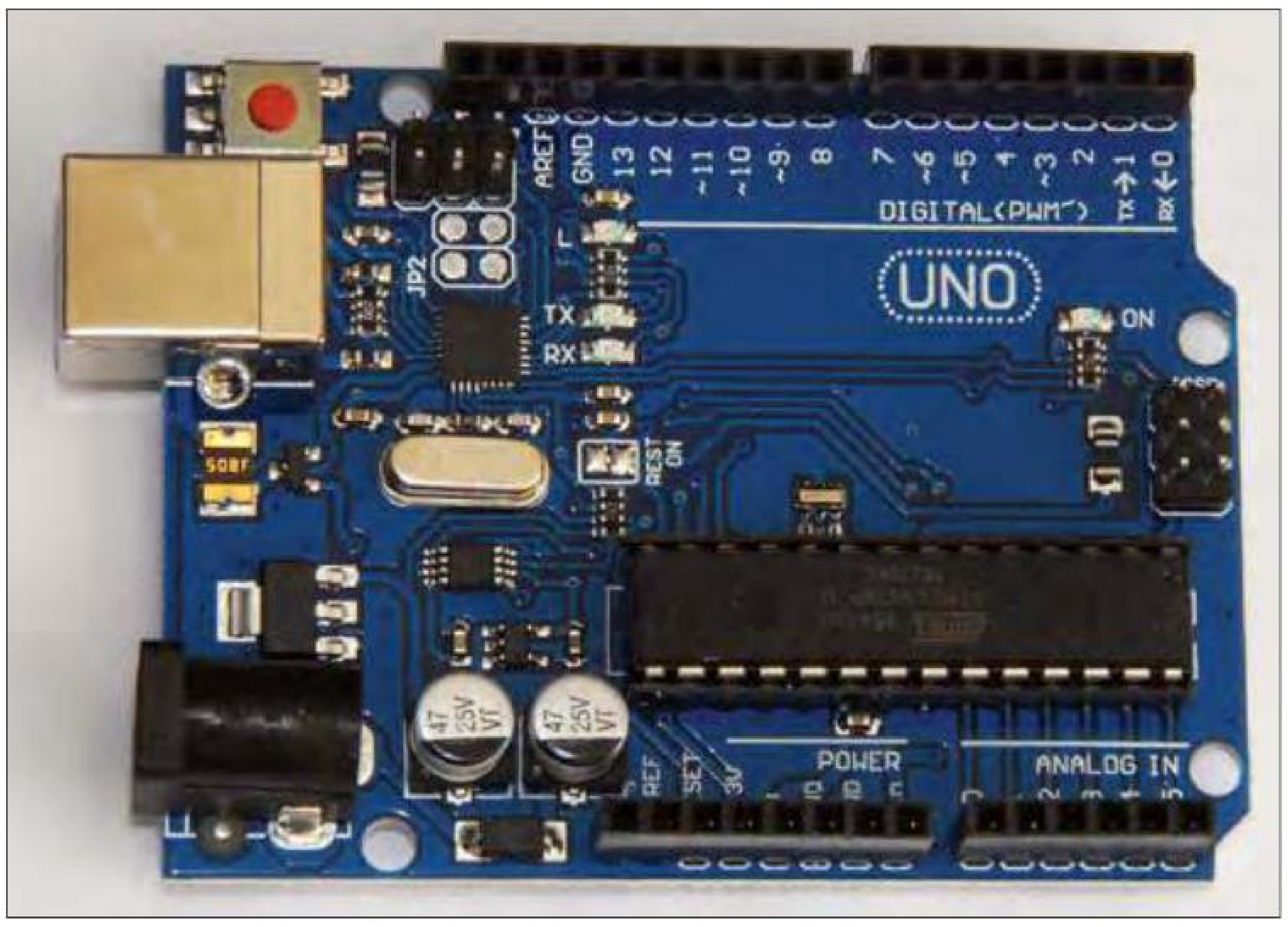


Fig.1. An Arduino Uno board – they're incredibly cheap, and with the use of XOD visual programming software, now very easy to program. You can have projects up and running in literally minutes.

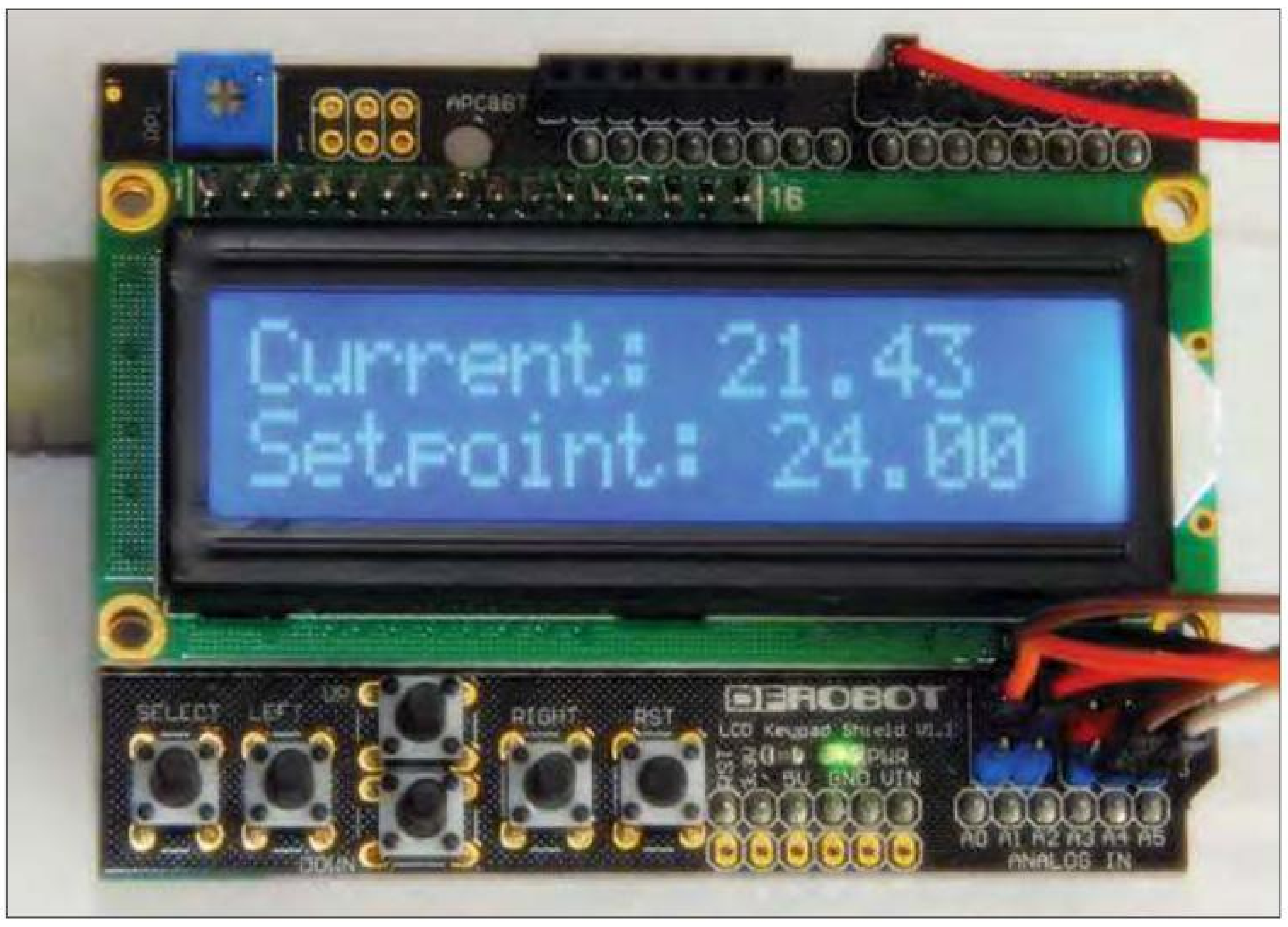


Fig.3. The DFRobot 1602 LCD Keypad Shield module plugged into the Arduino Uno board. The Uno is programmed with the XOD software discussed in this article and the display is showing the current temperature and the setpoint at which the alarm LED lights.

Note that to use the software, the Arduino board must be connected to the USB port of the PC – the USB connection powers the board. (There is also an online simulation function available for XOD, but I have found it to be a bit erratic.)

Using the software

The first step after downloading XOD is to do the in-built tutorial. It's quite detailed, and if you do all the suggested activities (the tutorials are done in the actual programming environment), it will take some hours. That said, after the first ten lessons, you'll have enough information to start programming. You can return to the tutorial at any time by going to: Help > Open Tutorial Project. I also found this page helpful: **https://dronebotworkshop.com/getting-started-with-xod/**

The hardware

In this example, I'll use:

- Arduino Uno, for example the board shown in Fig.1
- Microchip MCP9700 temperature sensor IC
- LED with suitable dropping resistor
- 10kΩ potentiometer
- DFRobot 1602 LCD Keypad Shield module (Fig.3).

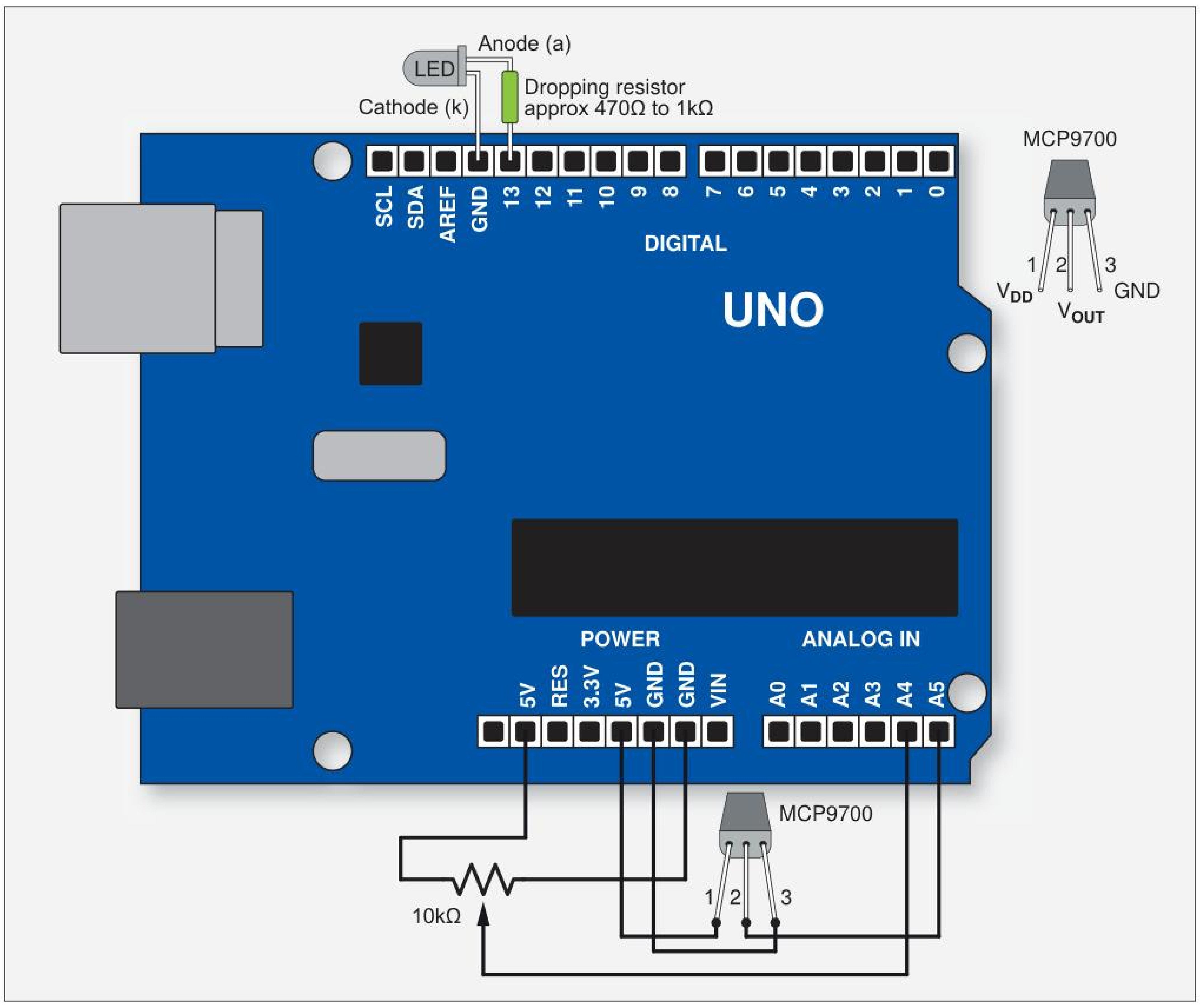


Fig.2. The initial set up for the Arduino board temperature alarm using an MCP9700 sensor and LED with dropping (current-limiting) resistor. (LCD shield not yet added.)

Let's start with the temperature sensor. Connect the appropriate pins (see Fig.2) from the sensor to the on-board 5V and ground, and the output to port A5 (analogue 5). (Note: all connections are labelled on the board.)

Connect the LED between pins D13 and ground. (To speed things up, I buy LEDs pre-wired with a resistor for 12V use. They're still plenty bright enough, even when working off 5V.) Ensure correct polarity.

Finally, connect the pot across the 5V and ground terminals, with its wiper (middle terminal) connected to port A4. You should now have what is shown in Fig.2. (We will get to the LCD shield later.)

Writing a program

Select File > New Project – you will be then confronted with a largely blank screen. This is your working area where you will build the program. On the far left and far right are two narrow columns – these display information that helps you as you proceed.

Let's make a temperature alarm, and to make it trickier, let's average the temperature reading over about five seconds and condition the sensor output so we're working in degrees Celsius (°C). We'll then add a pot input to adjust the setpoint temperature, and then an LCD to show both setpoint and real-time temperatures. Sound hard? It isn't!

Functional building blocks

You can see from the pictures in this article that the program is built from functional blocks (called 'nodes') – but where are they? Double clicking in the main work area will bring up a 'search nodes' box. We want to start the programming by having an analogue

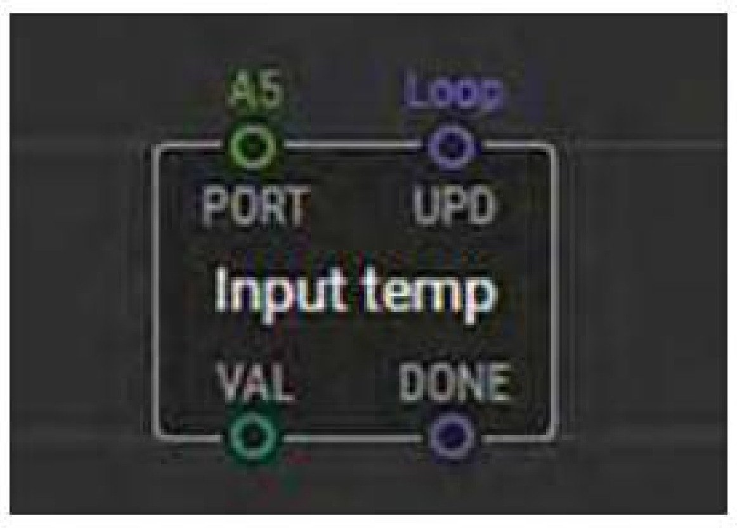

A5
Loop
PORT
UPD
Input temp
VAL
DONE

Fig.4. Adding the first XOD block – an 'analog-sensor' functional box.

sensor (the temperature sensor) input to the Arduino board. Typing 'analog' into the search box brings up `xod/common-hardware/analog-sensor`. Select this by double clicking on it. A functional box called `analog-sensor` will then be added to the main work area – see Fig.4.

Clicking on this box brings up information in the two screen columns. At right, each label on the functional box is explained, while at left these can be user-configured. The labels on this first box are:

`PORT` – the analogue port that is going to be used for this input on the Arduino board. Change it to A5.
`UPD` – how the input is updated. The options are 'never', 'on boot' and 'continuously'. Set it to 'continuously' – shown on the box as 'loop'.
`VALUE` – the output value, configured automatically to be in the range 0-1.
`DONE` – outputs a pulse when the reading is done (we don't use this).

We can re-label the box to read something other than `analog-sensor` – and now, using the column at left, label it `Input temp`.

So we've now added an analogue input on port `A5` that is continuously updated and is labelled in a way we can immediately understand.

Creating an input

But is our input actually reading anything? Double-click the work area and type `debug` and a series of options pops up – select `xod/debug/watch` and connect this block to the output (`VAL`) of the `Input temp` block. Upload the program to the board by pressing the 'bug' symbol at bottom right of the screen. This loads the program in debug form – you can then see the real-time working of the program, as shown in Fig.5.

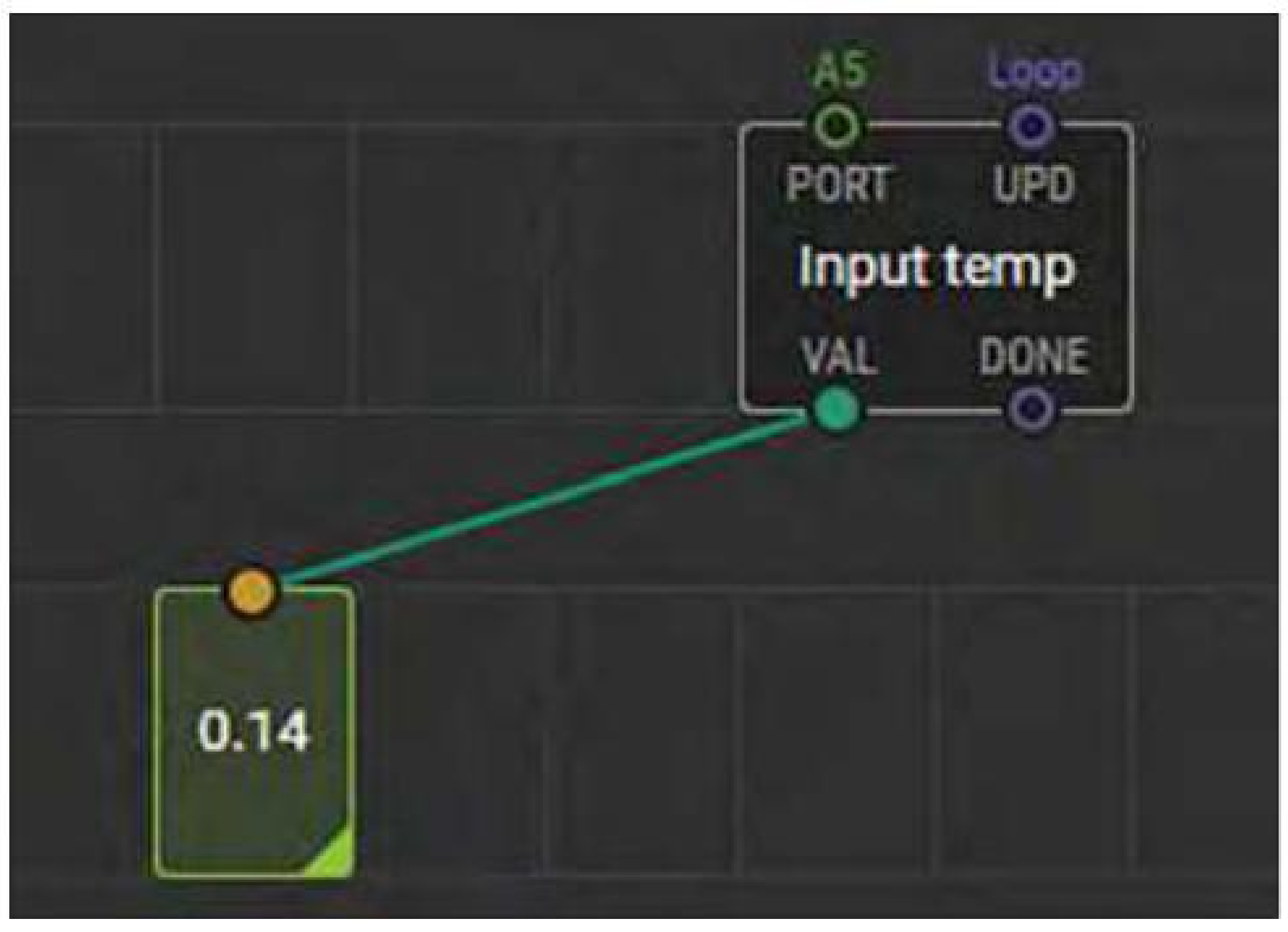

A5
Loop
PORT
UPD
Input temp
VAL
DONE
0.14

Fig.5. Adding a `watch` box and connecting it to the `Input temp` block.

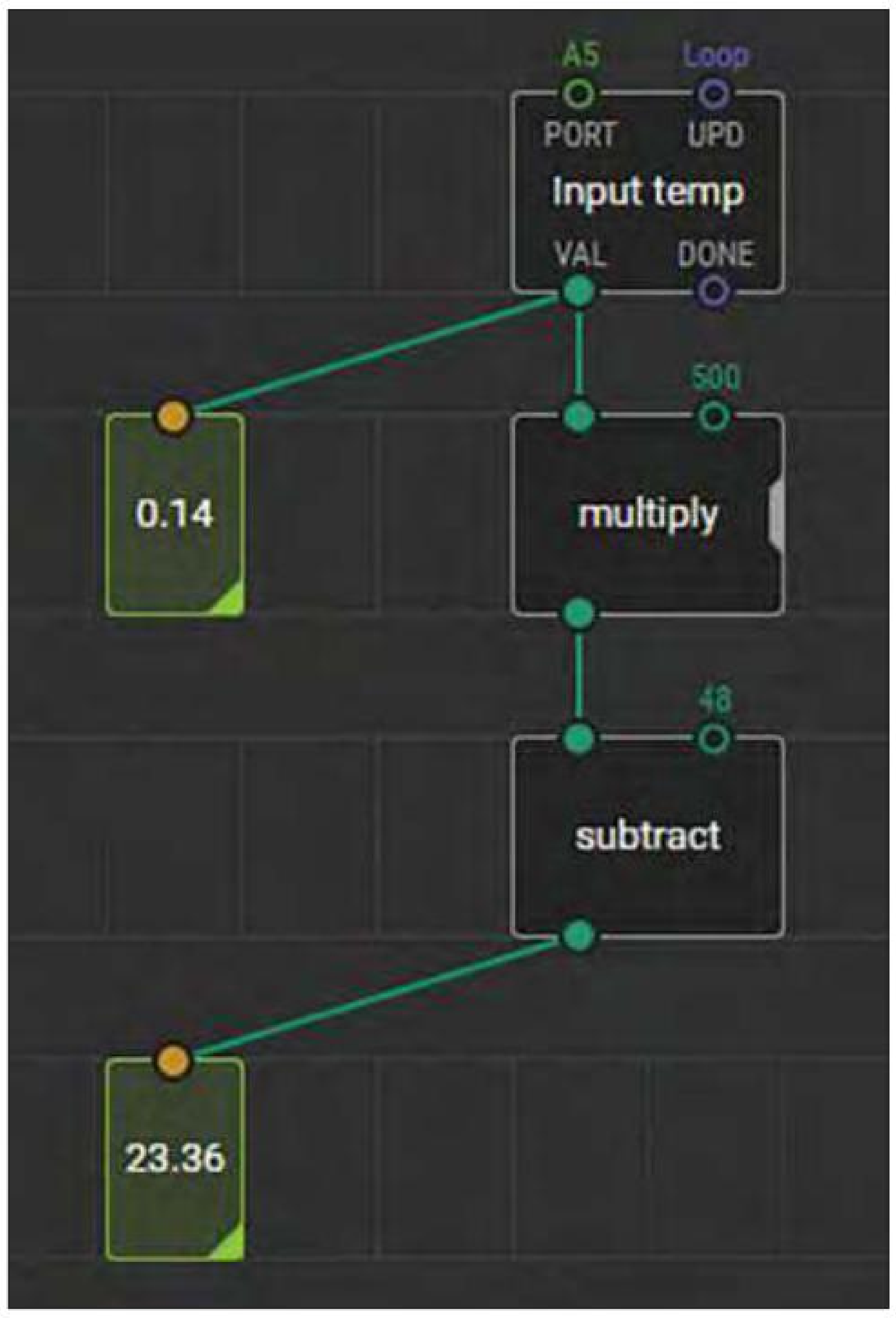

A5
Loop
PORT
UPD
Input temp
VAL
DONE
0.14
500
multiply
48
subtract
23.36

Fig.6. Adding blocks to display the sensed temperature in °C.

By heating and cooling the temperature sensor you should now see the value in the `watch` box changing – we have an input!

Hmm, but there is also an issue – at room temperature, the output of the sensor is showing `0.14` – not a 'real' temperature reading. We need to convert the reading to degrees Celsius, and with this sensor that means multiplying this reading by 500 and subtracting 50. So how do we do that?

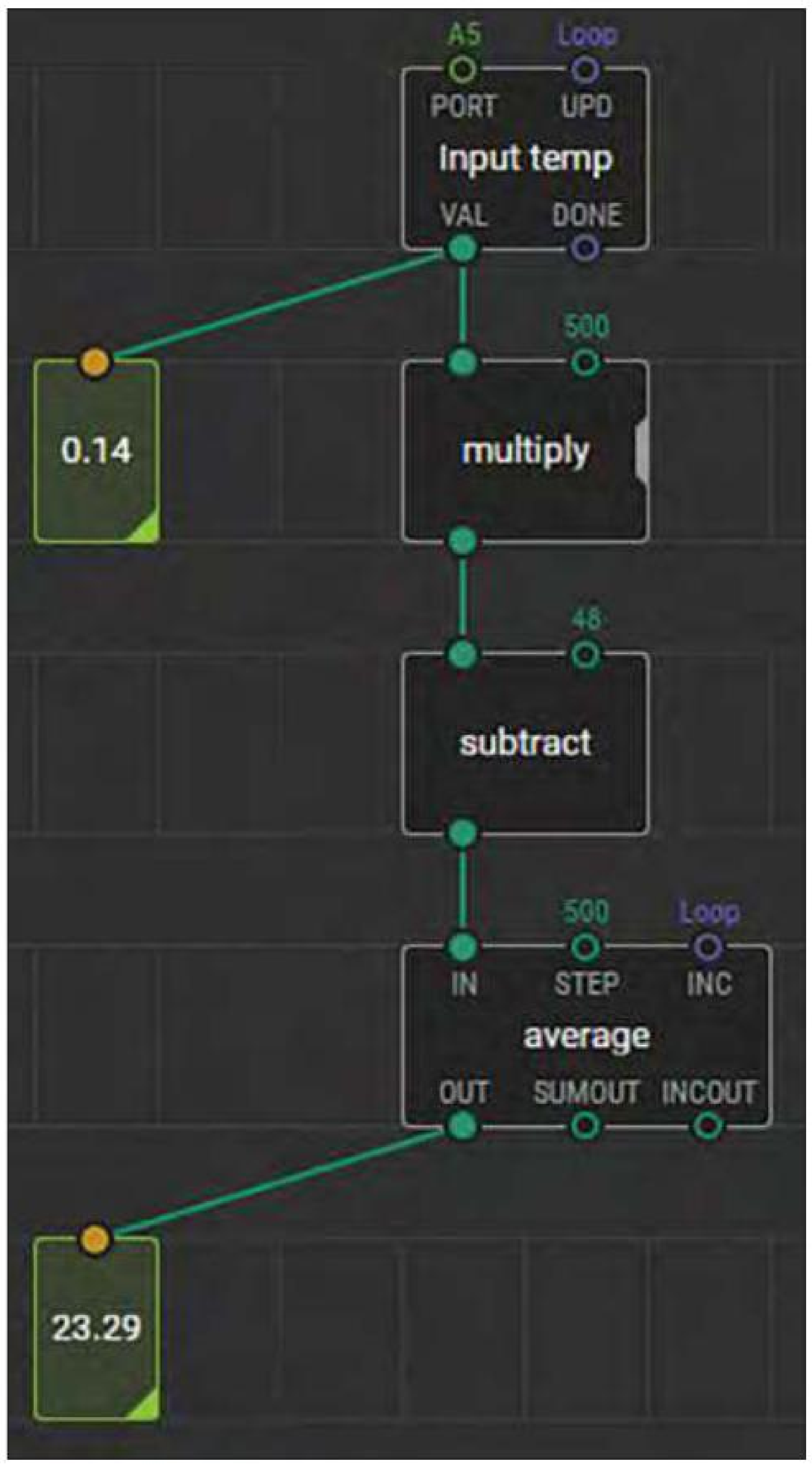

A5
Loop
PORT
UPD
Input temp
VAL
DONE
0.14
500
multiply
48
subtract
500
Loop
IN
STEP
INC
average
OUT
SUMOUT
INCOUT
23.29

Fig.7. Using a user-generated `average` function to generate five-second averages of the temperature readings.

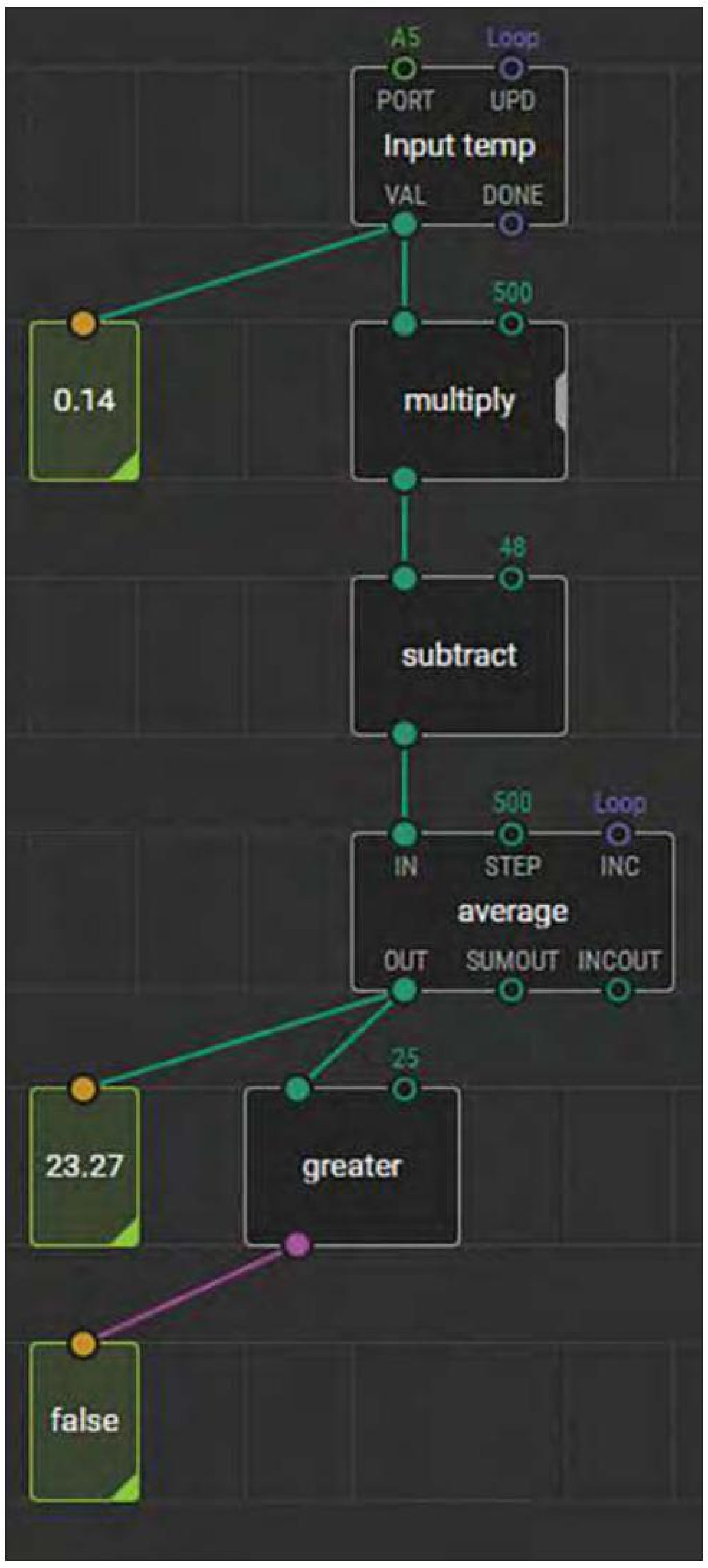

A5
Loop
PORT
UPD
Input temp
VAL
DONE
0.14
500
multiply
48
subtract
500
Loop
IN
STEP
INC
average
OUT
SUMOUT
INCOUT
23.27
25
greater
false

Fig.8. A `greater` box enables the system alarm to be added. The `true/false` box lets you see if the alarm is activated.

Processing the input values

Double-clicking in the workspace and searching under 'multiply' brings up `xod/core/multiply` that we then select. We connect the `IN1` of the `multiply` box to the `VAL` output of the input temperature box, and using the left-hand column on the screen, change the `IN2` value to `500`.

We then add a `xod/core/subtract` box and subtract `50`. (Well, as you can see here, I actually used `48` – that offset giving better accuracy with the particular temp sensor I was using.)

We can then add another `watch` box to the output of the `subtract` box and upload the program in debug mode to the board. You should then see the reading showing in degrees Celsius, as shown in Fig.6.

Data processing

Now we want to average the temperature reading; eg, over five seconds. But if you double-click on the work area and start typing 'average', you'll find you don't get any result – there is no 'averaging' block available as standard.

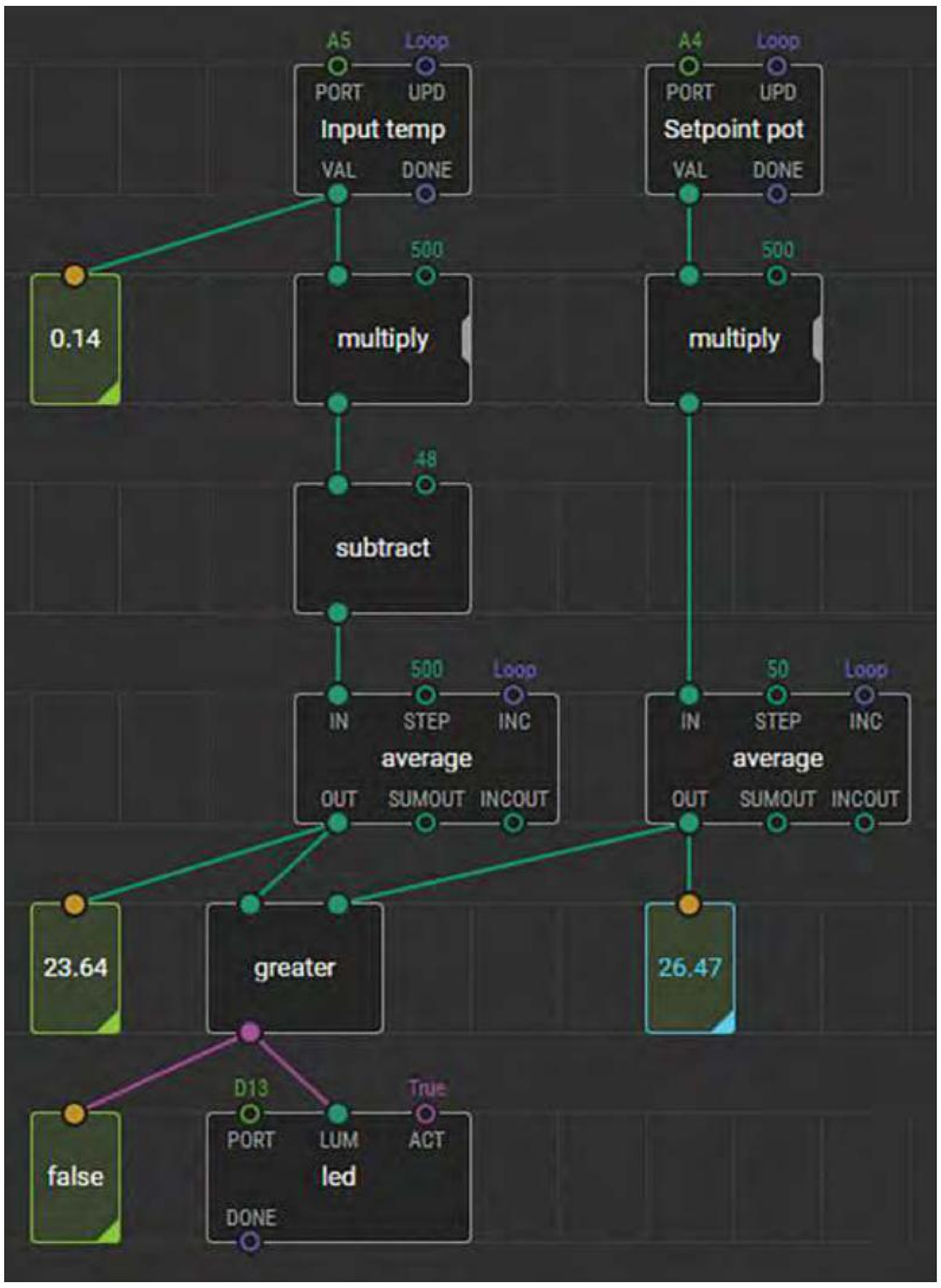


Fig.9. Here we add blocks to: a) enable the Arduino to light the LED when the alarm is triggered; and b) use the pot to adjust the setpoint.

However, one of the advantages of the XOD system is that users can upload their own useful blocks for others to use. These can be found on the XOD website under 'libraries'. By going to File > Add Library and typing in the library name as shown on the website, these extra blocks can be added. `average` is one of them – it's under 'gst/average/average'. This box has six labels, but we're interested in only three – `in`, `out` and `step`.

Click and drag a line from the output value of the `subtract` box to `IN` on the `average` block. Change `STEP` to `500` and `INC` to `Continuously` (shown on the box as `LOOP`). Now we're averaging the output of our temperature sensor over about five seconds. But will it work? Add a `watch` box to the output of the `average` box, upload the program in debug mode to the board, and then check that the temperature reading is now averaged over a period. (Fig.7)

Adding the alarm

Now we'll add the `alarm` part of the program. Select a `xod/core/greater` box and link `IN1` to the output of the `average` box. Change `IN2` to `25` (as in, 25°C). Add a `watch` box to the output of the `greater` box. (See Fig.8.)

Upload the program in debug form and check that the `watch` box changes from `false` to `true` when the temperature of the sensor is over 25°C. (Note how the watch box can show both values (numbers) and logic (true/false) states.).

Driving the LED

The next step is to light an LED when the alarm is triggered. Add a `xod/common-hardware/led` box to the screen – searching under 'led' will bring this box up. Change its port to `D13` and connect its `LUM` pin to the output of the `greater` box. (See lower left, Fig.9.)

Upload the program in debug mode. Now the LED lights when the measured temperature is above 25°C, averaged over about five seconds. (You'll also notice that an on-board LED lights too – it's also connected to `D13`.)

External pot input

In the program's current format, the set-point temperature of the alarm is typed into the software. But what if we wanted to adjust the setpoint with the external pot? As shown here, doing this takes only moments. Note that we are using port `A4` for the pot input, and again, we multiply its output by `500` (no subtraction necessary) and then average the output. Here the set-point is 26.47°C. (Would you like to round this number? If so, find the `xod/math/round` box and add it.) See right-hand side of Fig.9.

Connecting an LCD display

Now, what about showing on an LCD screen both the setpoint and monitored temperature? But I must be honest. There is a potential hardware hurdle here – and that is that LCD modules have a wide variety of different pinouts. The default LCD 16 × 2 programming block in the XOD software was not a match for the DFRobot 1602 LCD Keypad Shield module I used – that is, it wasn't 'plug-in compatible' so I had to make some port changes. Doing this meant looking up the data sheet for the LCD module and changing the designated ports on the XOD LCD block (covered in a minute) to match the functions of the different pins. In the case of the DFRobot LCD this comprised:

LCD pin	**Arduino port**
RS	D8
EN	D9
D4	D4
D5	D5
D6	D6
D7	D7

To add the LCD shield, first remove the existing temperature sensor, LED and pot connections. Then plug in the LCD shield, ensuring it is positioned as far towards the end of the Arduino board as possible. With this done, we now need to reconnect the pot and temperature sensor inputs using the `analog inputs` pins on the front of the shield. The LED output also needs to be moved. The port numbers remain the same, so this takes only moments – although you will need to change from male to female connectors.

The `text-lcd-parallel-16x2` box is then added. `L1` refers to line 1, and `L2` to line 2 on the display; each is a separately addressable (see Fig.10). Line 1 is then linked to the averaged input temperature via a `xod/core/concat` box that is used to display the word '`Current`:' and the numerical temperature reading. Line 2 uses a similar `concat` box to display the word `Setpoint:` and the pot's averaged output. The changed program is then uploaded in debug mode, and you should see the complete system working. (If the LCD does not show anything, try adjusting the on-board contrast pot.)

Conclusion

And there we have it! We have built a digital over-temperature warning system that lights an LED when the setpoint temperature is exceeded and displays on an LCD the current temperature and the setpoint temperature. Yes, it's a basic system, but it needed no code and was quick and easy to put together.

Now consider how easily it could be extended or altered – for example:

- Changing the temperature averaging time requires altering only one number in the software
- Fine-tuning the accuracy of the temperature sensor is as easy as changing the offset number
- Adding hysteresis requires just one more programming box (a hysteresis block is available in the Libraries)
- Altering what is displayed on the LCD (eg, displaying the temperature averaging time rather than setpoint) is as easy as drawing a new connecting line and changing the text label
- Comparing the difference between two temperatures requires only the addition of another temperature sensor and a few boxes of programming.

And the XOD software? The proof of the pudding is in the eating – in just this story we've not only introduced the programming software, but also managed to build a complete project!

What next?

I hope this piece inspired you to try XOD with an Arduino – there will be plenty more XOD-based projects in upcoming issues to tempt newbies into discovering the power of digital control.

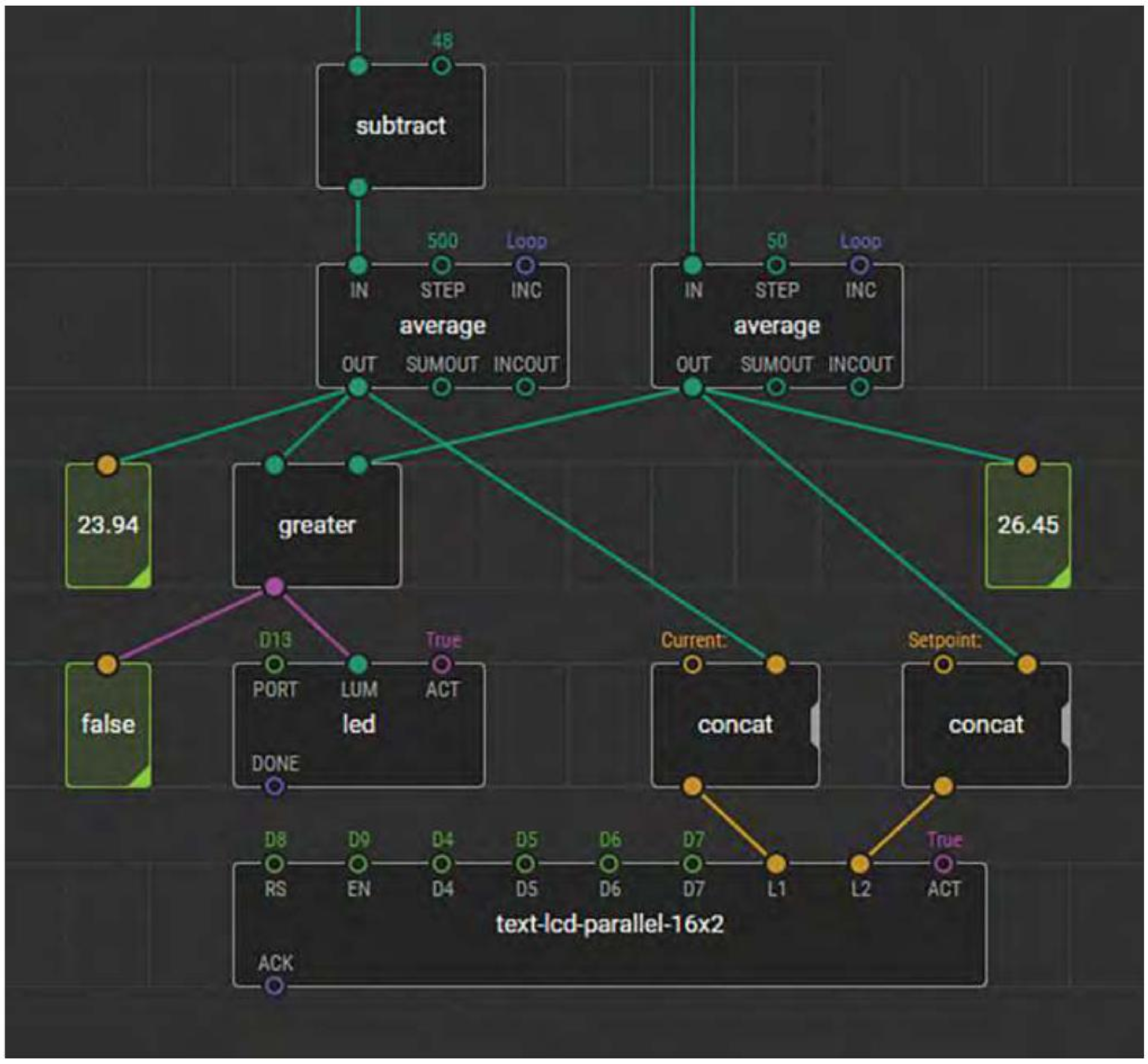


Fig.10. Adding the DFRobot 1602 LCD Keypad Shield module completes the alarm project. (Note: this image only shows the lower section of the blocks; the top section is identical to the top two rows in Fig.9.) Pin compatibility issues meant this was the only complex stage of the project – but XOD took it in its stride, and the LCD shield works well – see Fig.3.

XOD files

The XOD file discussed in this article can be downloaded from the March 2020 page of the *PE* website.

Teach-In 8 CD-ROM
Exploring the Arduino

This CD-ROM version of the exciting and popular *Teach-In 8* series has been designed for electronics enthusiasts who want to get to grips with the inexpensive, immensely popular Arduino microcontroller, as well as coding enthusiasts who want to explore hardware and interfacing. *Teach-In 8* provides a one-stop source of ideas and practical information.

The Arduino offers a remarkably effective platform for developing a huge variety of projects; from operating a set of Christmas tree lights to remotely controlling a robotic vehicle wirelessly or via the Internet. *Teach-In 8* is based around a series of practical projects with plenty of information for customisation. The projects can be combined together in many different ways in order to build more complex systems that can be used to solve a wide variety of home automation and environmental monitoring problems. The series includes topics such as RF technology, wireless networking and remote web access.

PLUS: PICs and the PICkit 3 – A beginners guide
The CD-ROM also includes a bonus – an extra 12-part series based around the popular PIC microcontroller, explaining how to build PIC-based systems.

SOFTWARE
The CD-ROM contains the software for both the *Teach-In 8* and PICkit 3 series.

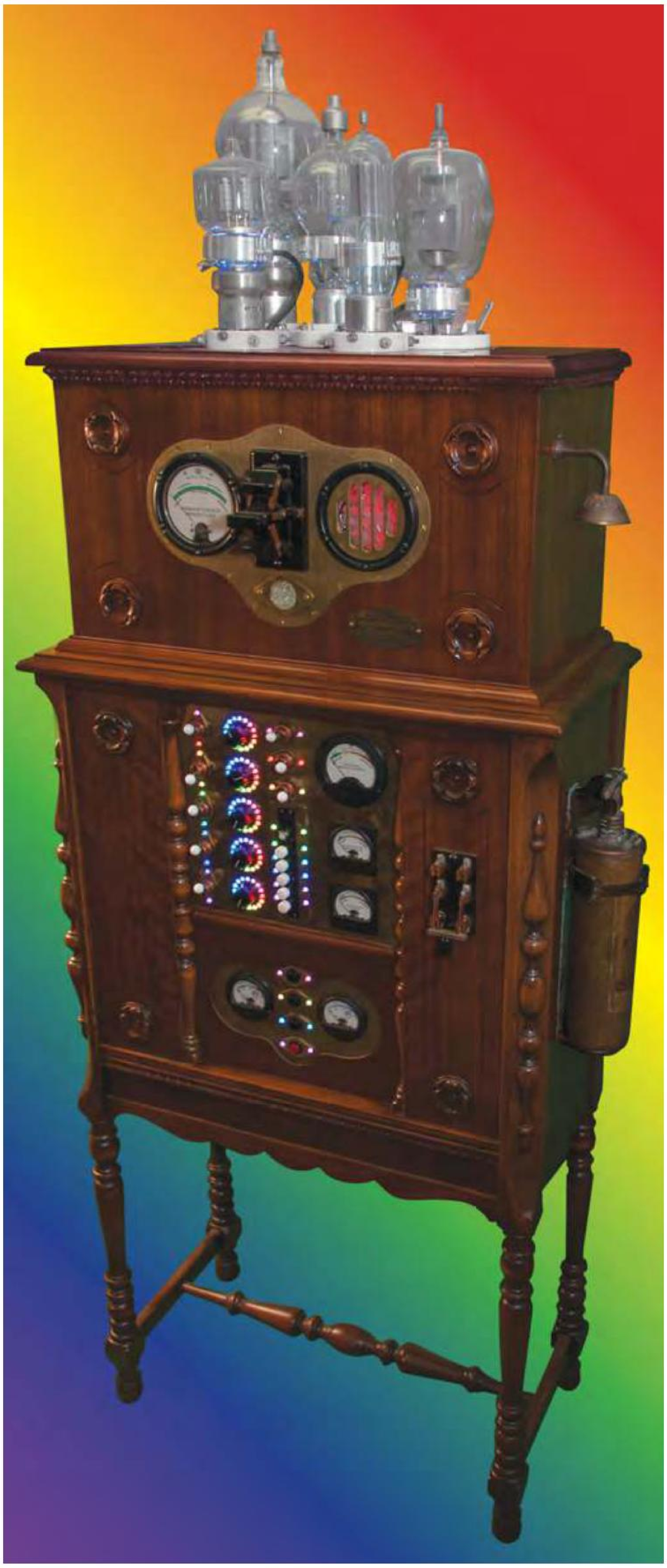
PRICE
£8.99
Includes P&P to UK if ordered direct from us

ORDER YOUR COPY TODAY!
JUST CALL 01202 880299 OR VISIT www.epemag.com

Max's Cool Beans

By Max the Magnificent

Flashing LEDs and drooling engineers



Prognostication Engine.

I've said it before, and I'll say it again, 'Show me a flashing LED, and I'll show you a man drooling.' The simple truth is that you cannot have too many LEDs in your hobby projects or electronic systems.

For example, take my *Inamorata Prognostication Engine* (Fig.1.), whose task it is to predict the mood of my wife (*Gina the Gorgeous*) as I'm leaving my office and heading for home. (By an ironic twist of fate, should Gina ever discern the *Engine's* purpose, I won't need it to predict her mood of the moment.) In addition to the tricolour LEDs illuminating the vacuum tubes on top and powering the furnace in the upper cabinet, the main control panel is festooned with the little rascals: two for each toggle switch and pushbutton switch, and 16 for each potentiometer.

At the moment, I'm just using these LEDs to present a rainbow effect. Now that I know everything works as planned, I'm poised to start programming them in earnest, but this has caused me to reflect on how we might enhance our projects using only a single unicoloured LED, and then – of course – working our way up to grander things.

A single unicolour LED

Just saying 'single unicolour LED' brings a little tear to my eye, but I'm trying to be brave, and I bet that with a little thought we can make even this interesting. Let's start by considering a single-pole, single-throw (SPST) toggle switch that we're using to control something or other.

Note that we are going to be playing with a very ordinary, standard, individually packaged LED that's distinct from the switch; that is, we aren't considering a component with an integrated LED looking like a halo surrounding the center of the switch, for example.

The first thing is to consider is whether the switching action is orientated vertically or horizontally (both options are used in the *Inamorata Prognostication Engine*); also, if our LED is to be mounted above, below, left, or right of the switch (Fig.2).

There are two reasons I'm portraying the LED as green in these images. The first is that the human eye is most sensitive to a yellowish-green colour. The second is that the original LED I plucked out of my treasure chest of spare parts when I set to writing this column happened to be green (such

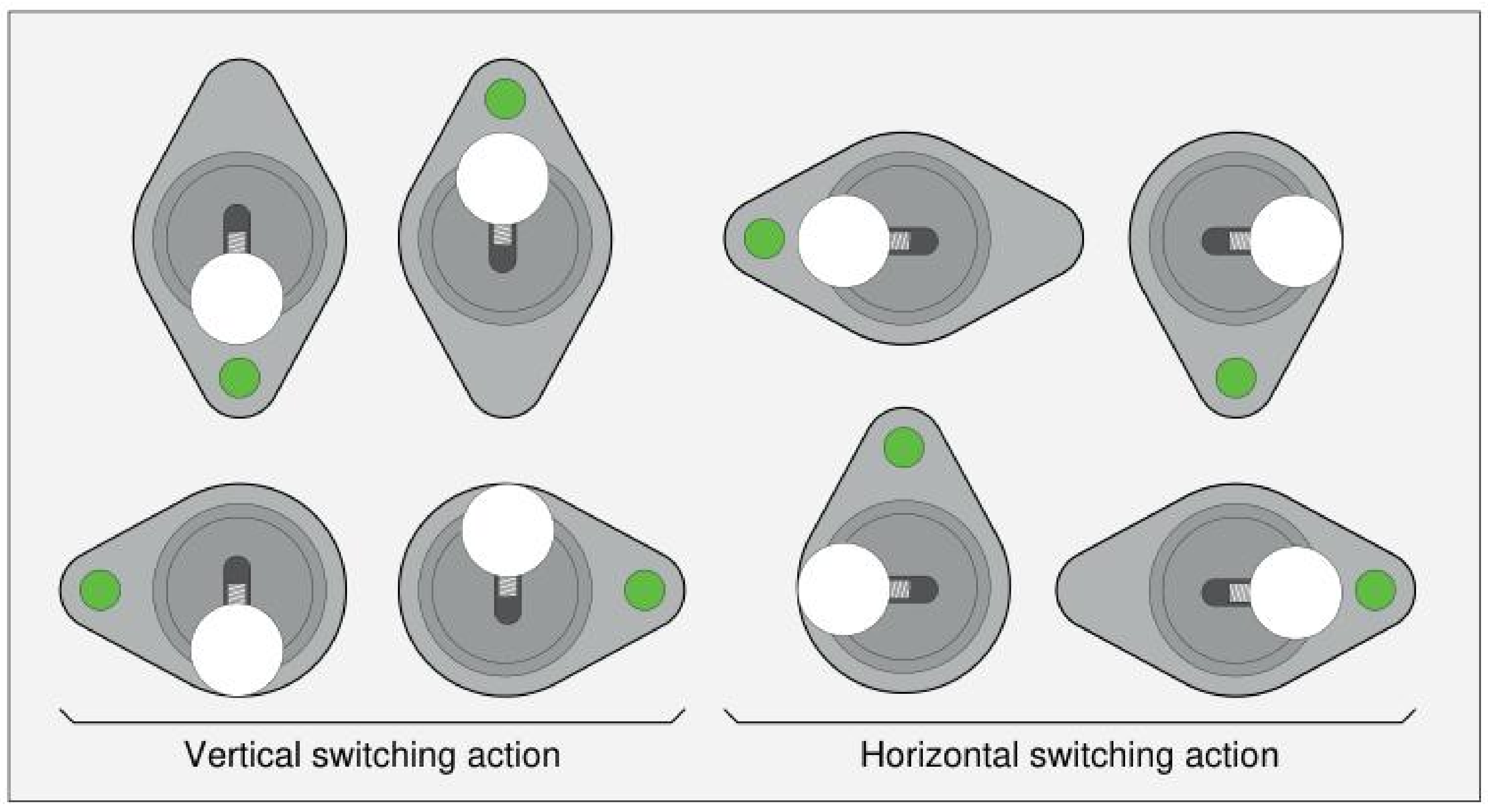


Fig.2. Alternative switch orientations and LED locations.

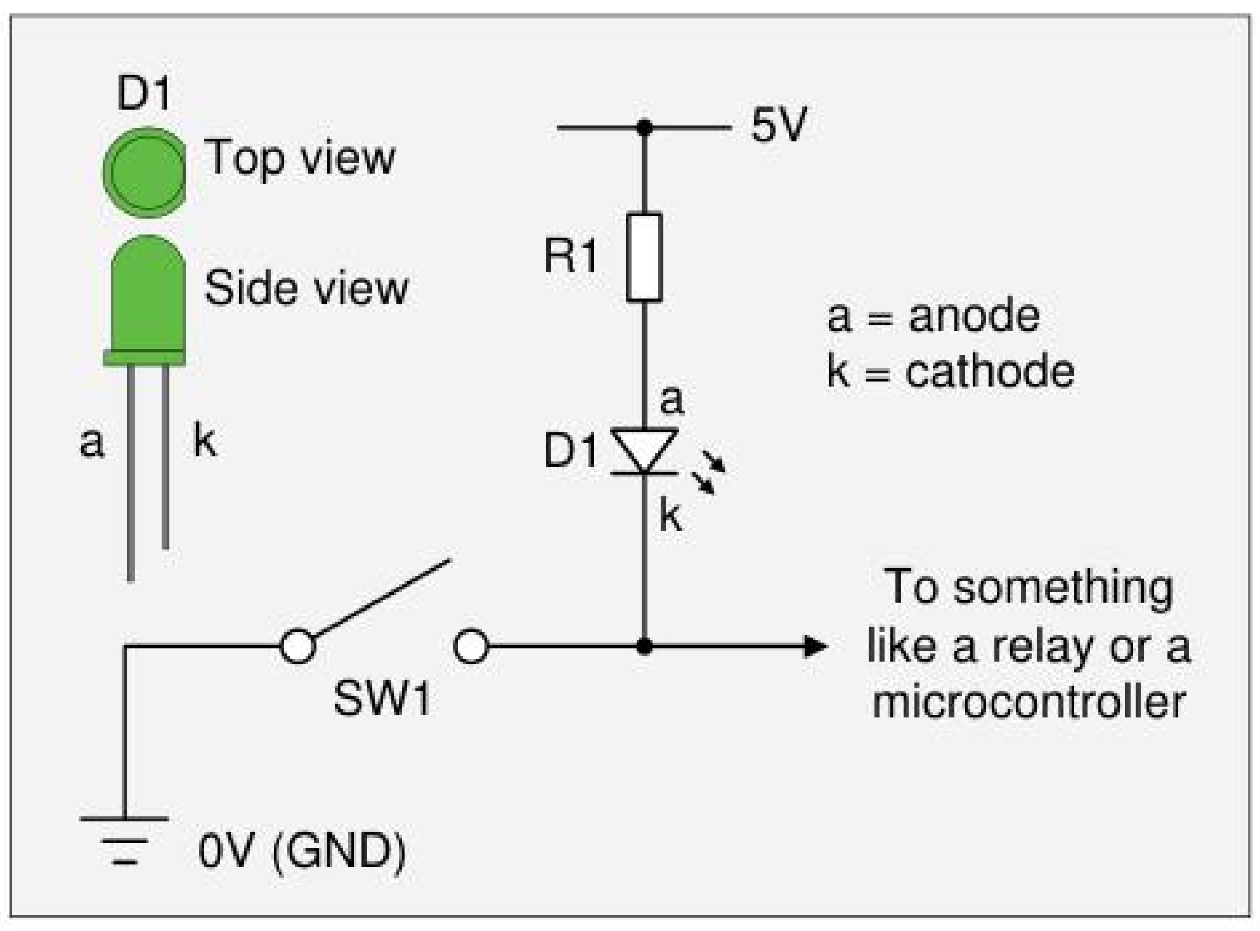


Fig.3. A switch driving a LED directly.

are the unpredictable whims of fate).

Whichever switch-orientation/LED-location combo we choose, let's start by considering the switch driving the LED directly without the aid of a microcontroller (Fig.3). Note that the reason I'm showing a 5V supply here is that I'm planning on using an Arduino Uno in later experiments. Also note that the LED's positive terminal, the anode (a), is the longer lead, while its negative terminal, the cathode (k), is indicated by the flat side on the plastic package.

Just to remind ourselves as to how these things work, and to ensure we're all tap dancing to the same drumbeat, two of the parameters on an LED's data sheet are its forward voltage drop (V_f) and its maximum forward current (I_f). Let's assume that the V_f and I_f for our LED are 2V and 20mA (0.02A), respectively.

Using Ohm's law, we know that $V = IR$, which means that $R = V/I$. Since our power supply is 5V and our diode drops 2V, this means that our resistor R1 = (5 – 2) / 0.02 = 3/0.02 = 150Ω.

For the purposes of these discussions, we will assume our switch is orientated so as to have a vertical switching action and that the LED is mounted below the switch. We will also assume that if the lever on the switch is pointing towards the LED, then we will regard the switch as being in its On/Closed/Active state; contrawise, if the lever is pointing away from the LED, we will regard the switch as being in its Off/Open/Inactive state (Fig.4).

This brings us to an interesting point, which is that in the UK one typically presses a toggle switch down to make it active and up to make it inactive. In the case of rocker switches, one presses the bottom of the rocker to activate the switch and the top to deactivate it. (One exception to this rule is in aviation, where toggling the switch up turns it on.) By comparison, things are usually the other way around in the US, and you take your chances in the rest of the world. You can do what you want in the case of your own projects, but consistency is probably more important than orientation, and it's always a good idea to make your systems 'local friendly.'

Adding an Arduino

We're going to have to agree on some form of terminology to help us keep track of things as we proceed. Let's start with the following to describe the previous example of the switch driving the LED directly:

Switch -> On; LED -> On
Switch -> Off; LED -> Off

While this is functional, it's hardly inspiring. Is there any way we can add a little pizazz? Well yes, and the easiest way to do so is by bringing a microcontroller into the picture. For the purposes of these discussions we'll use an Arduino Uno. Furthermore, we need to plan ahead, because as the sophistication of our effects increases, switch bounce may become an issue.

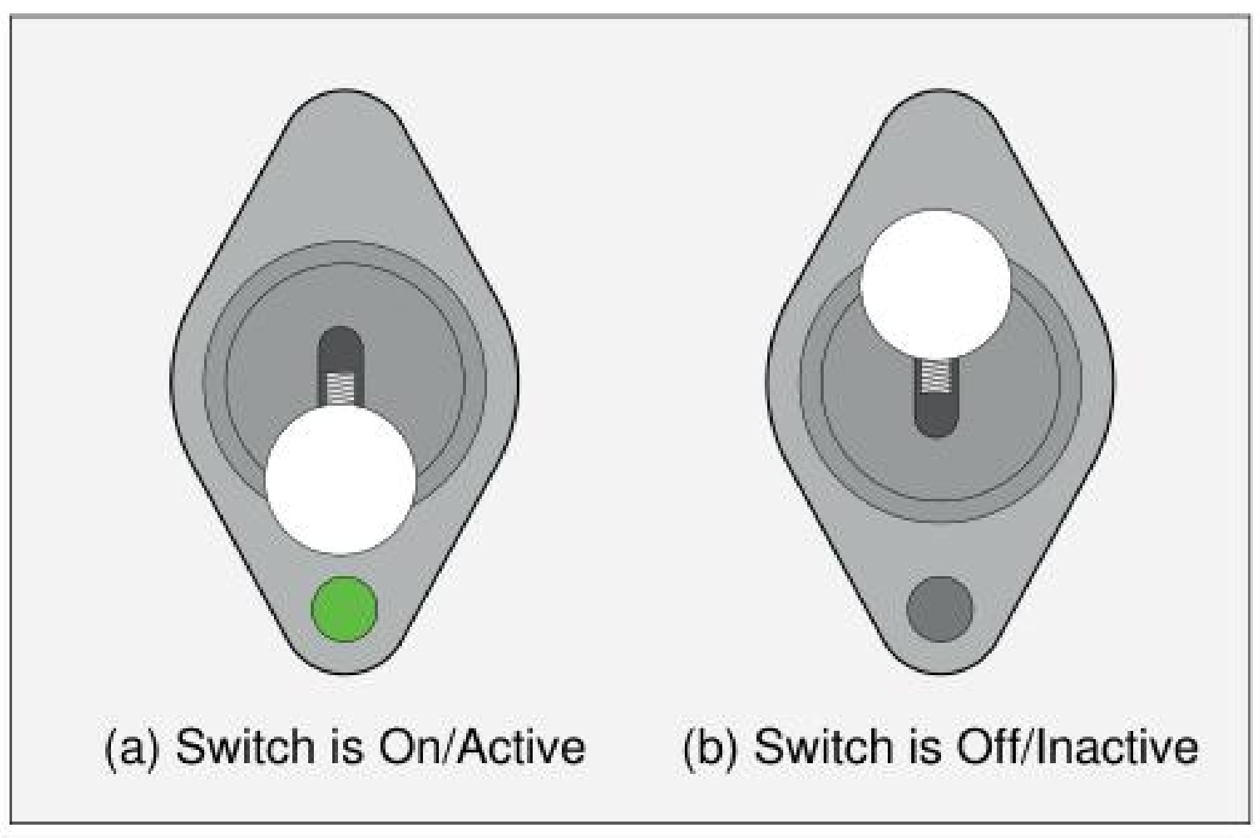

(a) Switch is On/Active (b) Switch is Off/Inactive

Fig.4. Our switch's active/inactive states.

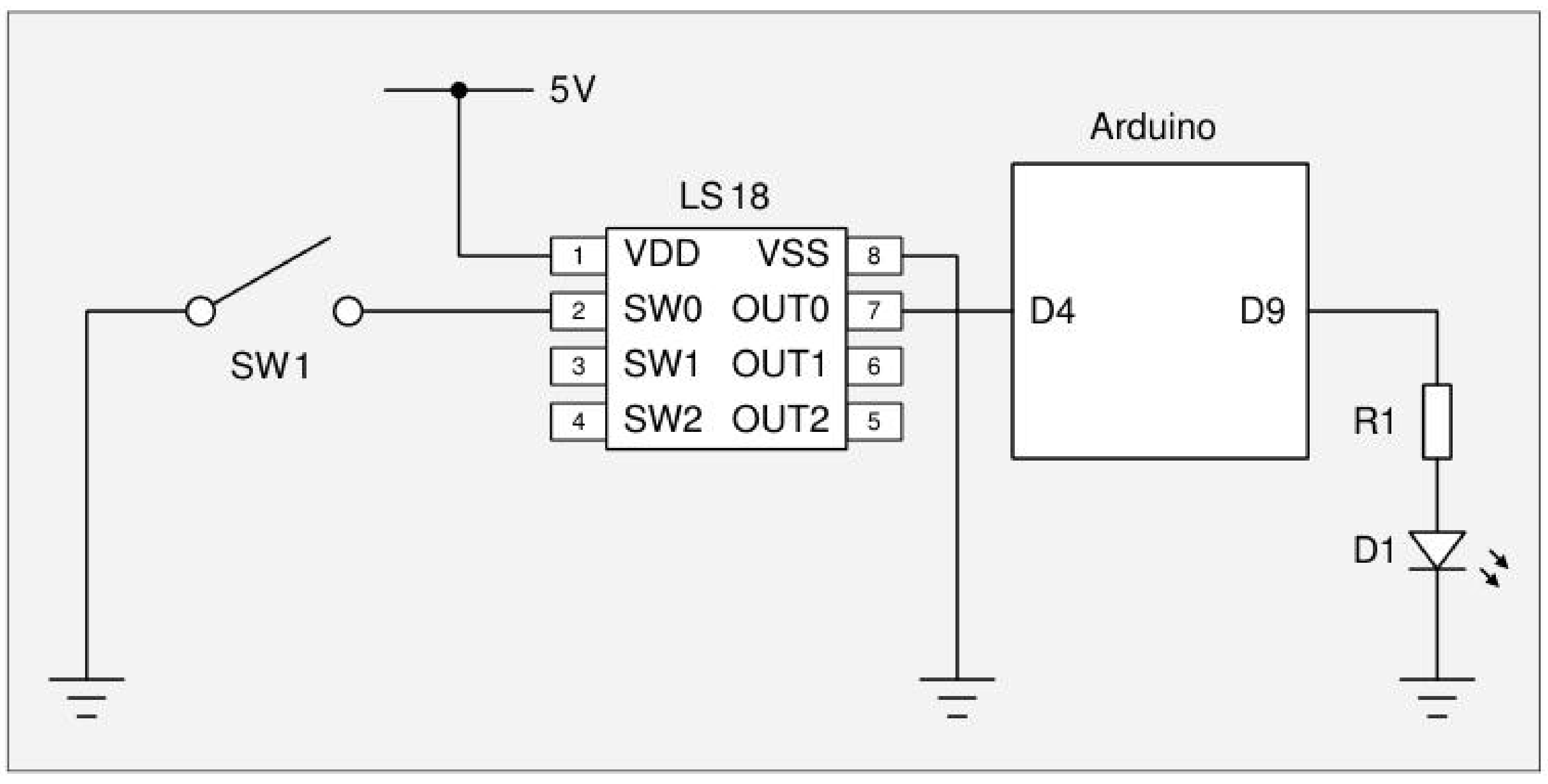


Fig.5. Using an LS18 to remove switch bounce and an Arduino to control our LED.

As a reminder, when we turn a switch on or off, it can actually bounce on-off-on-off… multiple times before settling in its new state. This applies to toggle switches, rocker switches, pushbutton switches, etc. This isn't a problem in the case of a light switch in a house, for example, because any flickering of the light that may ensue happens far too quickly for our eyes and brains to perceive it. However, it's a different matter when you have microprocessors that can sample their inputs millions of times a second, because they could see what we think of as a single switch transition as comprising multiple events.

There is a wide variety of hardware and/or software techniques we can use to debounce our switches. In this case, I'm using an LS18 dual in-line (DIL) chip from LogiSwitch.net (Fig.5). This is a 3-channel device, which means it can handle three switches. Excluding noise spikes on its inputs, which it rejects, the outputs from the LS18 follow their respective inputs, delayed by 20ms after the final bounce (Fig.6).

The 'D4' and 'D9' annotations in Fig.4 represent the Arduino's digital input/output (I/O) pins 4 and 9, respectively. Observe that we aren't using pull-up resistors on the LS18's SW0 input or the Arduino's D4 input because the LS18 takes care of this for us (I love this chip, and I now use it – and its LS118 cousin – in all of my projects).

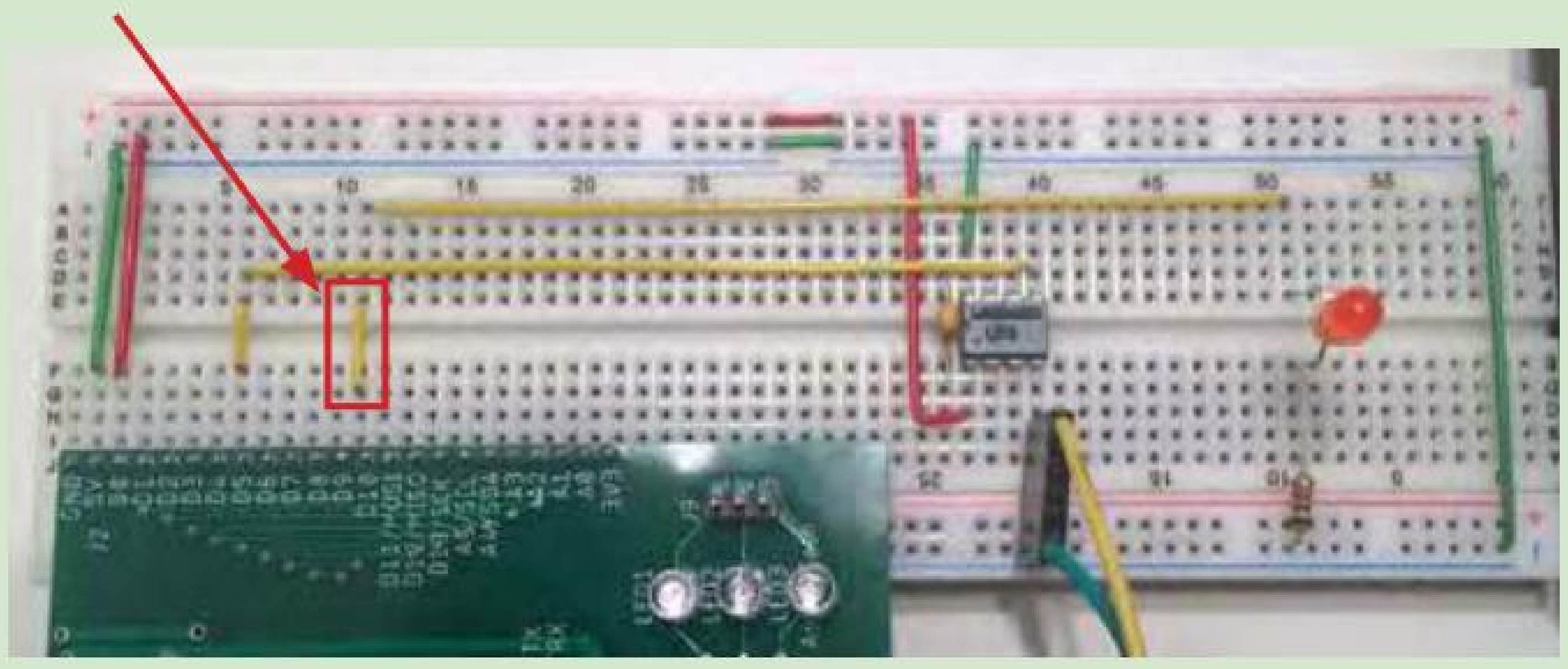
A word to the wise

This is just a little bit of advice from me to you. It doesn't matter how good you think you are or how simple is the circuit that you are building, your life will be a lot less frustrating if you do the following:

1) Draw your circuit diagram on a piece of paper.
2) Compare your physical circuit to the circuit diagram and mark off the wires one-by-one.

When I was wiring up the breadboard for the first program, I forgot to include one little jumper wire. If I'd followed my own advice above, I would have saved myself a lot of hassle. As it was, I spent an inordinate amount of time tracking this little scamp down. That's a very simple example – just imagine troubleshooting Fig.3 from last month. Yes, I know, you'd rather not, but following my advice will help you stay sane(-ish).

Forgetting to include this wire caused a lot of frustration!



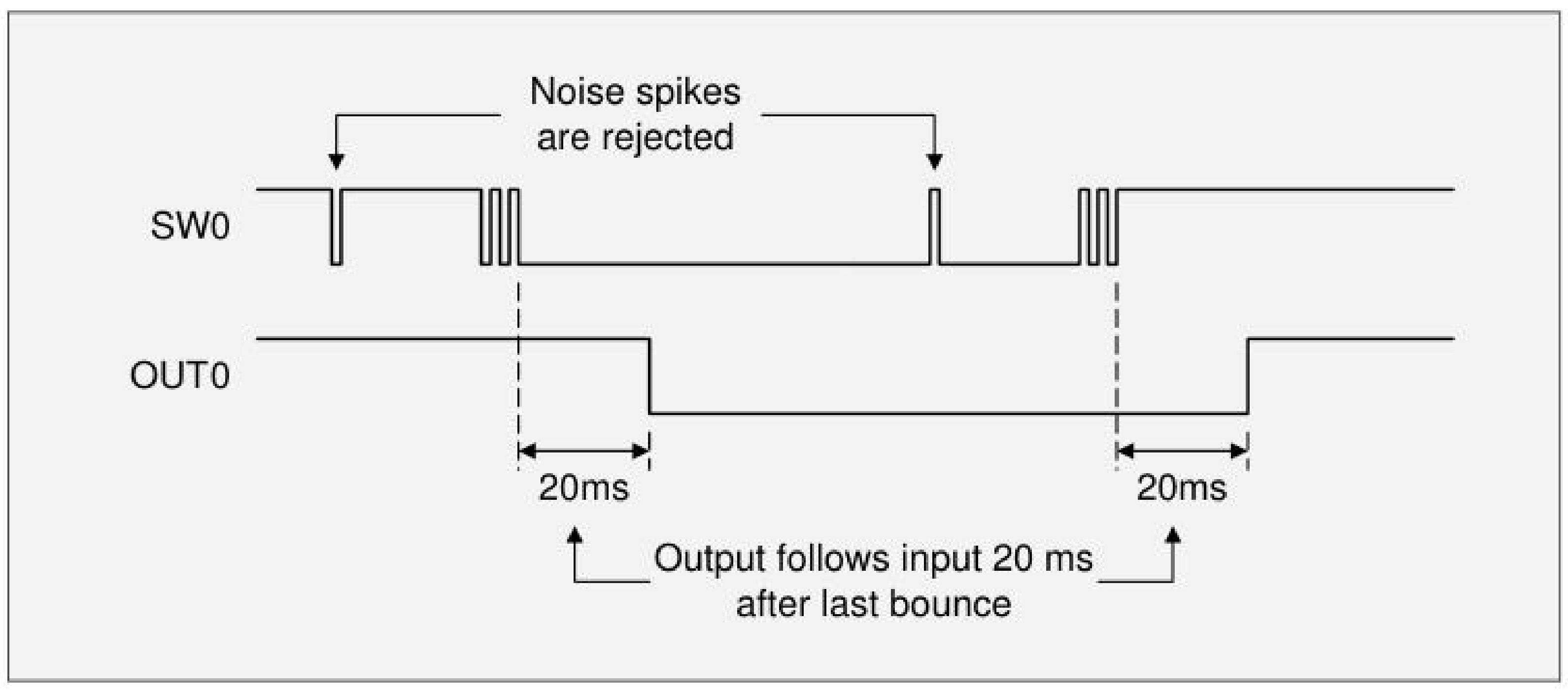

Noise spikes are rejected
SW0
OUT0
20ms
20ms
Output follows input 20 ms after last bounce

Fig.6. Operation of the LS18 switch debouncer chip.

For your delectation and delight, I will make any code I create for the Arduino as part of this series of articles available for you to download, peruse, and ponder. Just to provide a point of reference, I've created an Arduino sketch (program) that replicates the effects of the switch driving the LED directly (**https://bit.ly/2tfezZ0**).

```
#define SWITCH_ON      LOW
#define SWITCH_OFF    HIGH

#define LED_ON        HIGH
#define LED_OFF        LOW

int PinSwitch = 4;
int PinLed    = 9;

int OldSwitchState;
int NewSwitchState;

void setup ()
{
    pinMode(PinSwitch, INPUT);
    pinMode(PinLed,   OUTPUT);

    OldSwitchState = digitalRead(PinSwitch);

    if (OldSwitchState == SWITCH_ON)
    {
        digitalWrite(PinLed, LED_ON);
    }
    else
    {
        digitalWrite(PinLed, LED_OFF);
    }
}


void loop ()
{
    NewSwitchState = digitalRead(PinSwitch);

    if (NewSwitchState != OldSwitchState)
    {
        if (NewSwitchState == SWITCH_ON)
        {
            digitalWrite(PinLed, LED_ON);
        }
        else
        {
            digitalWrite(PinLed, LED_OFF);
        }

        OldSwitchState = NewSwitchState;
    }
}
```

I fear that when you look at my program, you might think the way I've done things is 'overkill.' For example, I ended up using around 50 lines of code, but I could have written this first sketch much more concisely as follows:

```
int PinSwitch = 4;
int PinLed    = 9;

void setup() {
  pinMode(PinSwitch, INPUT);
  pinMode(PinLed,   OUTPUT);
}

void loop() {
  if (digitalRead(PinSwitch) == LOW)
    digitalWrite(PinLed, HIGH);
    else digitalWrite(PinLed, LOW);
}
```

Actually, we could remove the `if()` statement and reduce the body of our `loop()` function to a single line if we were feeling particularly bold (email me if you aren't sure how to do this). There are several reasons why I created the first sketch the way I did, the most important being to set the scene for what is to come, like having a special start-up sequence when power is first applied to the system, for example.

By now you must think I'm a coding genius – my good manners prevents me from contradicting you. I have learnt a few tricks along the way, and if you are new to all of this then at the end of the article I've put together a few little tips, hints and handy rules of thumb that may make your life a little easier.

Pretentious? Moi?

You may call me pretentious if you will (although the radiance of my mother's smile will no longer fall upon you), but I feel that simply turning our LED hard on and hard off lacks a little *je ne sais quoi*.

The thing is, now that we have added a microcontroller into the picture, we can do all sorts of capriciously cunning things. 'The world is our oyster,' as they say (or octopus, or any mollusc of your choice). For example, when the switch closes, we could flash the LED three times before pausing and then going hard on (**https://bit.ly/2FbmWqY**):

Switch -> On; LED -> Flash-Flash-Flash-Pause-On
Switch -> Off; LED -> Off

If you look at my code, you'll see I'm using calls to the delay() function. I'm doing this to make the code as easy to understand as possible. In practice, however, using this function would not be a good idea in this situation. This is because `delay()` is a blocking function, which means it prevents the Arduino from doing anything else (apart from servicing interrupts). Happily, there are better ways of doing things, but they're a bit more complicated so we'll leave them as a topic for another day.

Returning to our discussions, another possibility is that – instead of turning it hard on and off – we fade the LED up and down, respectively:

Switch -> On; LED -> Fade-On
Switch -> Off; LED -> Fade-Off

So, how do we perform this fade? It is possible to dim an LED by lowering its supply voltage, but this typically doesn't work as well as one might hope, not the least because things will go pear-shaped once your supply voltage approaches the LED's forward voltage-drop value. The alternative is to turn the LED

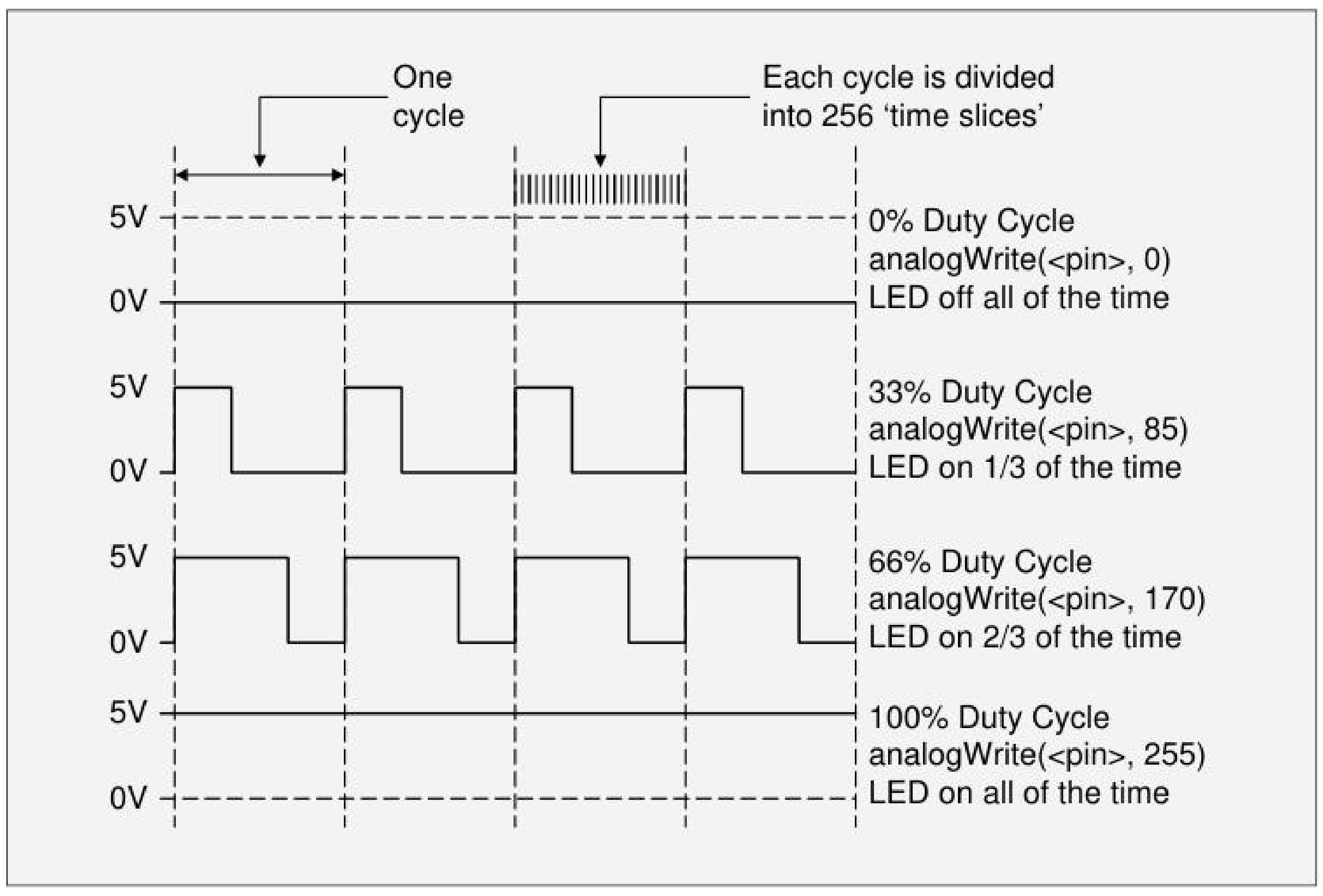


Fig.7. Using PWM to control the brightness of an LED.

fully on and fully off very quickly. If the LED is only on for 50% of the time, for example, then it will only appear to be half as bright as if it were on for 100% of the time.

Fortunately, microcontrollers can turn things on and off very quickly indeed, thereby allowing us to implement something we call pulse-width modulation (PWM). In the case of the Arduino, we have a function called `analogWrite()` that allows us to specify a pin number and a PWM parameter, where this parameter is an 8-bit unsigned integer that can represent values between 0 (fully off) and 255 (fully on) – see Fig.7.

In the case of the Arduino, each complete cycle shown in Fig.7 consumes only around 1/500th of a second. Using this technique, the LED is turned on and off so quickly that no flicker is perceptible to the human eye. The Arduino Uno has six pins that support PWM via the `analogWrite()` function: 3, 5, 6, 9, 10, and 11 (thank goodness we decided to use pin 9 to drive our LED – it's almost as if we had a plan).

Let's commence with a 'clunky' approach where we 'fade up' by increasing the brightness in three big steps, starting at 0% and proceeding to 33%, then 66%, then 100%. We'll fade down in the same clunky way (**https://bit.ly/2u7GACa**). Once we've seen this in action, we'll modify our program to make the fades occur much more smoothly (**https://bit.ly/2MKPqfp**).

Feel the power!

Something else we might consider is the possibility of implementing a start-up sequence. That is, when power is first applied to the system, we want to do something a little bit special – perhaps something like the following (**https://bit.ly/2QGlGBQ**):

Power-up (Switch is on): LED -> ((flash-flash-flash-pause) * 3) -> On
Power-up (Switch is off): LED -> extended-flash -> Off

Now imagine a system that boasts a bunch of switches and LEDs, like the *Prognostication Engine*, for example. In this case, we can make our start-up sequence a feast for the eyes. Furthermore, since this bodacious beauty is equipped with tricolour LEDs, we might vary the entire colour scheme depending on its evaluation of the current state of play in the world (bright, cheerful colours when things are going well; more foreboding colours when the situation is turning ominous – DEFCON anyone? **https://en.wikipedia.org/wiki/DEFCON**).

Breathing, snoozing, and snoring

One more thing to consider is what happens if no one activates a switch for a certain period of time – say 10 minutes – which we might call the 'Inactive Timeout Period.' It might be nice to indicate this in some way.

There are two primary scenarios for when this period times out: either the switch and LED are currently on or they are currently off. Let's start with the case where they are on. One thing we could do would be to enter a mode where we give a really short periodic flash:

Inactive Timeout (Switch is on): LED -> Flash-Pause-Flash-Pause…
Inactive Timeout (Switch is off): LED -> Stays off

Another alternative is that the systems enters a 'breathing' mode (I've also heard this called 'snoozing' and 'snoring' on the basis that the system is sort of asleep) where the LED gradually fades up and down... and up and down:

Inactive Timeout (Switch is on): LED -> Breathing mode
Inactive Timeout (Switch is off): LED -> Stays off

Last but not least, if we are using a breathing mode when the switch is on, maybe we could use an intermittent flash when the switch is off:

Inactive Timeout (Switch is on): LED -> Breathing mode
Inactive Timeout (Switch is off): LED -> Flash-Pause-Flash-Pause…

When something occurs to wake the machine up again, maybe we could run the start-up sequence we discussed earlier.

There are a couple of reasons I haven't provided any code examples for these snoozing and snoring cases. The first is that we currently have only a single switch, so the only way to wake the system up would be to toggle our switch into its opposite state. If the switch were on when we entered the snoozing mode, for example, then the only way to wake the system would be to turn the switch off. Thus, implementing this mode would make a lot more sense if our system boasted a bunch of switches.

Another problem is our use of the `delay()` function, which makes it difficult for us to detect when some action occurs in the outside world and respond to that action in a timely manner. We will discuss techniques to work around this in a future column.

The excitement mounts!

On the off chance you're interested, I've created a video showing all of the effects we've discussed thus far in action (**https://bit.ly/2ZUvL1W**) – I'd love to hear what you think.

Also, remember that all of the effects presented above are just first-pass ideas that popped into my head as I was writing this column. It would be great if you could perform some experiments of your own and then email me if you come up with any cunning effects you would care to share. (Such generosity will be handsomely rewarded with fame, but no fortune – none.)

The really exciting thing is that, thus far, we've only considered a single toggle switch with a single unicolour LED. What about pushbutton switches (latching and momentary/non-latching)? What about bicolour and tricolour LEDs? And what about having two LEDs associated with each switch or pushbutton?

I don't know about you, but I'm becoming light-headed considering the multifarious potentials, prospects, and possibilities. I can't wait to see what we come up with in next month's column. In the meantime, I welcome your comments, questions, and suggestions. Until next time, have a good one!

Cool bean Max Maxfield (Hawaiian shirt, on the right) is emperor of all he surveys at **CliveMaxfield.com** – the go-to site for the latest and greatest in technological geekdom.

Comments or questions? Email Max at: **max@CliveMaxfield.com**

Max's Cool Beans cunning coding tips and tricks

I'll be providing downloadable Arduino sketches (programs) to accompany my next few *Cool Beans* columns, so I thought it might be a good idea to provide a rationale for the coding style I use.

The first point to note is that you use the Arduino's integrated development environment (IDE) to capture your sketches using the C/C++ programming languages. C++ is essentially C with extensions (like C on steroids). C++ is classed as an object-oriented language (OPP), but we really don't need to worry about what this means at this moment in time. What we think of as the 'Arduino Language' is a set of pre-defined C/C++ functions that someone has written for us and that can be called from our code.

The `main()` function

One thing that trips some people up if they already know C/C++ is that programs written in these languages always have top-level function called `main()`. In the case of the Arduino, however, there are two functions that are placed in your sketch by default, `setup()` and `loop()`, but no `main()` function. (You can also create your own functions, but your sketch must include the `setup()` and `loop()` functions.)

The IDE also adds a comment inside the `setup()` function saying, 'Put your setup code here, to run once.' Similarly, there's a comment inside the `loop()` function saying, 'Put your main code here, to run repeatedly' (their use of the word 'main' is unfortunate in this context).

Behind the scenes, when you click the 'Compile' icon, the IDE generates a temporary file that includes all of your code along with a `main()` function looking something like the following:

```
main ()
{
    setup();
    while(1) loop();
}
```

This calls the `setup()` function a single time, and then it calls the `loop()` function over and over and over again.

Cases and spaces

C/C++ are case-sensitive languages; for example, variables with names like `fred`, `Fred`, and `FRED` are treated by the compiler as being completely different entities.

C/C++ don't care whether you use whitespace characters or not (where whitespace characters include spaces, tabs, and blank lines). The main reason for using whitespace characters is to make your code more readable; they are stripped out when the code is compiled, so they won't make the final program that's loaded into the Arduino any bigger.

Speaking of tabs... don't use them. Yes, of course it's easier to press the `<tab>` key a single time rather than the `<space>` bar multiple times. The problem is that you don't know how many spaces other systems will associate with a tab, which may make your printouts look strange, for example. And let's not forget the 2017 article on the **StackOverflow.blog** website: *Developers Who Use Spaces Make More Money Than Those Who Use Tabs* (**https://bit.ly/37iB0eg**).

Comments

Anything on a line following two forward slashes (`//`) characters is seen as a comment. Multi-line comments can be started with a slash and a star (`/*`) and terminated with a star and a slash (`*/`). Comments really help you and others to understand and maintain your programs, so don't be miserly with these little scamps, but don't add comments when there's no need to do so (having a statement that says `a = a + 1;` followed by a comment that says: '`// Add 1 to a`' is the sign of someone who needs a little guidance). Use common sense.

Got style?

Do you remember the song *Style*, as sung by Frank Sinatra, Dean Martin, and Bing Crosby (**https://bit.ly/39qLpGG**)? There are a couple of lines that go: *You've either got or you haven't got style. (If you got it, you stand out a mile.)*. This is certainly true in the case of writing code.

A lot of coding style comes down to personal preference. If you work for a big company, you are expected to follow their in-house style. If you are doing this for yourself, you can develop your own way of doing things. The main thing is to be consistent so that you (and others) can more easily understand and maintain your code in the future.

I'm a hardware design engineer by trade. I'm self-taught regarding software, so you really shouldn't listen to anything I say regarding coding. On the other hand, I've been exposed to a lot of code and seen a lot of things that I like (along with a lot that I don't). My own style has evolved over the years as I've been exposed to different techniques. What follows is a subset of the coding techniques that work for me:

#define

I always use uppercase alphanumeric characters and underscores for my constant names; for example:

```
#define START_COUNT    0
#define END_COUNT    100
```

Rationale: This allows me to easily identify global constants in the body of the program.

Don't use 'magic numbers'

Take a look at the following statement:

```
for (i = 0; i < 10; i++)
{
    // Do stuff here
}
```

Both the `0` and the `10` would be considered to be 'magic numbers' because they appeared from nowhere as if by magic. Generally speaking, it's fine to use `0` (or `1`) because we all know that counts usually start or terminate at `0` (or `1`), but it's best to use a constant value for any other number; for example (assuming `MAX_COUNT` was previously declared using a `#define` statement):

```
for (i = 0; i < MAX_COUNT; i++)
{
    // Do stuff here
}
```

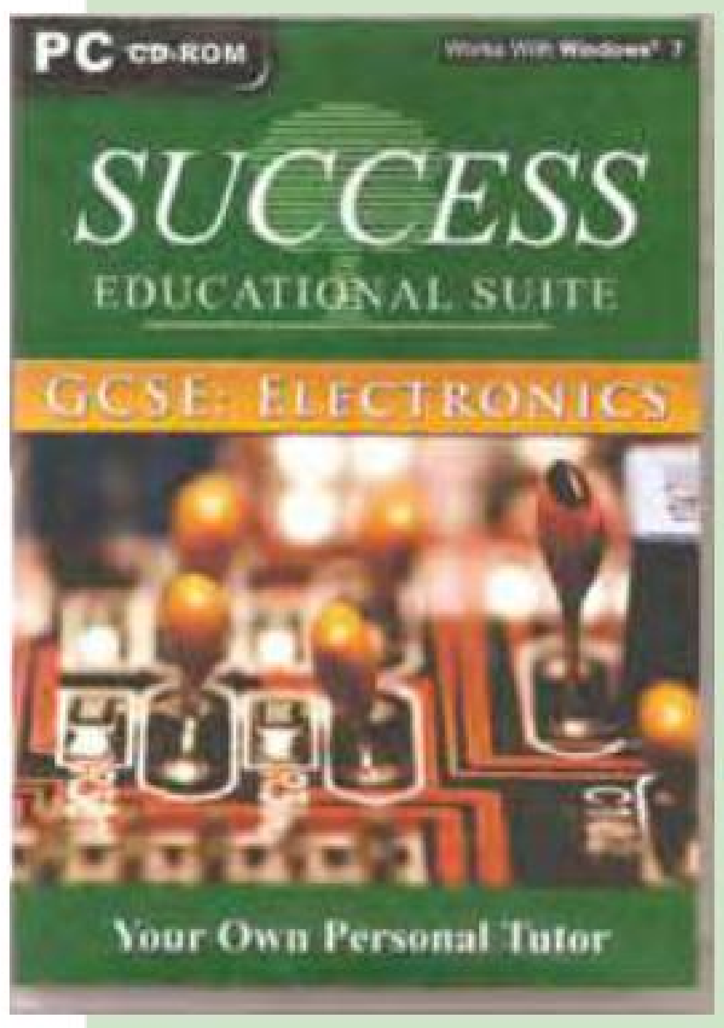
Next time

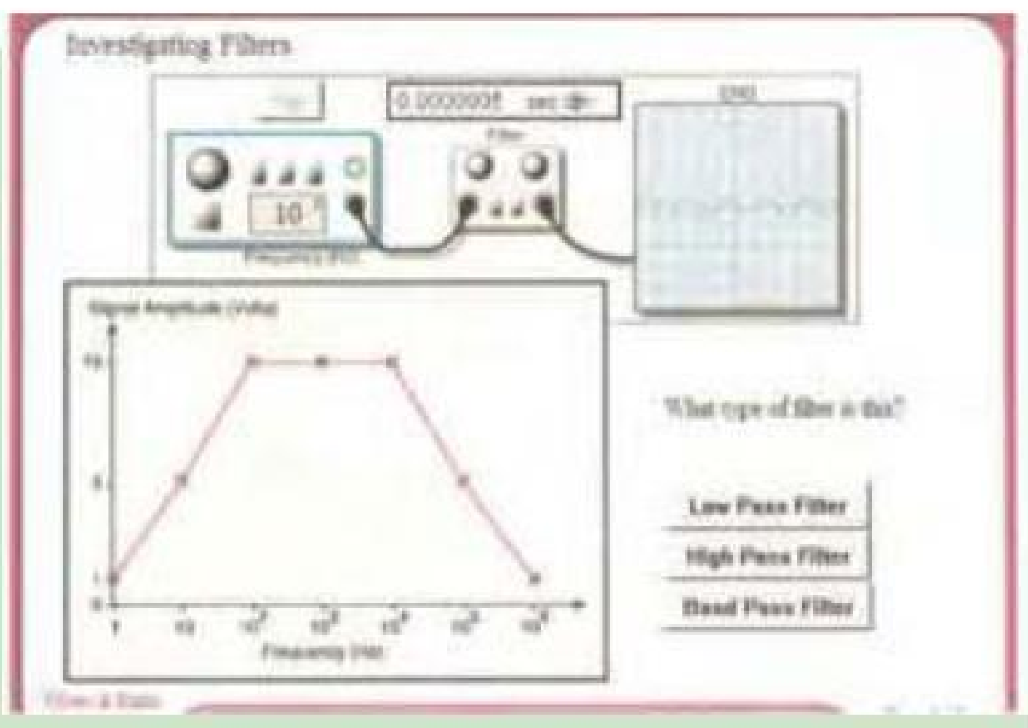
In my next column we'll look at variable names, function names, and function calls. In the meantime, if you have any coding tips and tricks you'd care to share, I'd love to hear about them.

PRACTICAL ELECTRONICS IS PLEASED TO OFFER YOU THESE

ELECTRONICS CD-ROMS

GCSE ELECTRONICS





Suitable for any student who is serious about studying and who wants to achieve the best grade possible. Each program's clear, patient and structured delivery will aid understanding of electronics and assist in developing a confident approach to answering GCSE questions. The CD-ROM will be invaluable to anyone studying electronics, not just GCSE students.

*** Contains comprehensive teaching material to cover the National Curriculum syllabus * Regular exercises reinforce the teaching points * Retains student interest with high quality animation and graphics * Stimulates learning through interactive exercises * Provides sample examination ques-tions with model solutions * Authored by practising teachers * Covers all UK examination board syllabuses * Caters for all levels of ability * Useful for self-tuition and revision**

SUBJECTS COVERED

Electric Circuits – Logic Gates – Capacitors & Inductors – Relays – Transistors – Electric Transducers – Operational Amplifiers – Radio Circuits – Test Instruments

Over 100 different sections under the above headings

£12.50
inc. VAT and P&P

CIRCUIT WIZARD

Circuit Wizard is a revolutionary software system that combines circuit design, PCB design, simulation and CAD/CAM manufacture in one complete package. Two versions are available – Standard or Professional.

By integrating the entire design process, Circuit Wizard provides you with all the tools necessary to produce an electronics project from start to finish – even including on-screen testing of the PCB prior to construction!

* Circuit diagram design with component library (500 components Standard,1500 components Professional) * Virtual instruments (4 Standard, 7 professional)* On-screen animation * Interactive circuit diagram simulation * True analogue/digital simulation * Simulation of component destruction * PCB Layout * Interactive PCB layout simulation * Automatic PCB routing * Gerber export * Multi-level zoom (25% to 1000%) * Multiple undo and redo * Copy and paste to other software * Multiple document support

This software can be used with the *Jump Start* and *Teach-In 2011* series (and the *Teach-In 4* book).

Standard **£61.25** inc. VAT. Professional **£75** plus VAT.

From £49.00

TINA Design Suite V12

FEATURED IN OUR TEACH-IN 2015 SERIES

Analogue, Digital, Symbolic, RF, MCU and Mixed-Mode Circuit Simulation and PCB Design with TINA

TINA Design Suite V12 is a powerful yet affordable software package for analysing, designing and real time testing analogue, digital, MCU, and mixed electronic circuits and their PCB layouts. You can also analyse RF, communication, optoelectronic circuits, test and debug microcontroller applications.

Enter and analyse any circuit up to 100 nodes (student), or 200 with the Basic (Hobbyist) version within minutes with TINA's easy-to-use schematic editor. Enhance your schematics by adding text and graphics. Choose components from the large library containing more than 10,000 manufacturer models. Analyse your circuit through more than 20 different analysis modes or with 10 high tech virtual instruments.

Present your results in TINA's sophisticated diagram windows, on virtual instruments, or in the live interactive mode where you can even edit your circuit during operation.

Customise presentations using TINA's advanced drawing tools to control text, fonts, axes, line width, colour and layout. You can create and print documents directly inside TINA or cut and paste your results into your favourite word procesing or DTP package.

TINA includes the following Virtual Instruments: Oscilloscope, Function Generator, Multimeter, Signal Analyser/Bode Plotter, Network Analyser, Spectrum Analyser, Logic Analyser, Digital Signal Generator, XY Recorder.

3D Viewer
File View Options

This offer gives you a CD-ROM – the software will need registering (FREE) with Designsoft (TINA), details are given within the package.

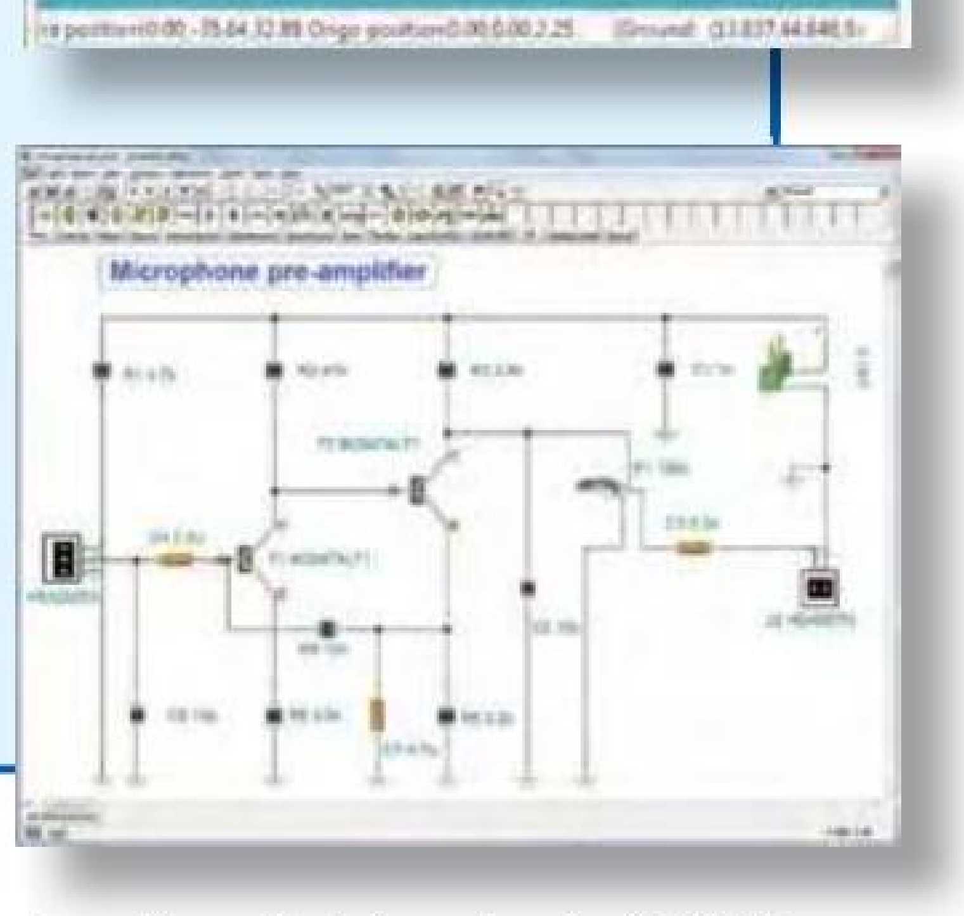
Get TINA 12 Design Suite (Hobbyist) for £129 or Student V12 version for £49

Prices include VAT and UK postage

+ get a 1 year free subscription for TINACloud the breakthrough cloud version of TINA which you can run on most operating systems and computers, including PCs, Macs, thin clients iPads and other tablets – even on many smart phones, smart TVs and e-book readers.

To order please either fill out and return the order form, or call us on 01202 880299

Alternatively you can order via our secure online shop at: www.epemag.com



PICmicro TUTORIALS AND PROGRAMMING

HARDWARE

PICmicro Multiprogrammer Board and Development Board

Suitable for use with the three software packages listed below

This flexible PICmicro microcontroller programmer board and combination board allows students and professional engineers to learn how to program PICmicro microcontrollers as well as program a range of 8, 18, 28 and 40 pin devices from the 12, 16 and 18 series PICmicro ranges. For those who want to learn, choose one or all of the packages below to use with the hardware.

- Makes it easier to develop PICmicro projects
- Supports low cost Flash-programmable PICmicro devices
- Fully featured integrated displays – 16 individual LEDs, quad 7-segment display and alphanumeric LCD display
- Supports PICmicro microcontrollers with A/D converters
- Fully protected expansion bus for project work
- USB programmable
- Compatible with the E-blocks range of accessories

£159.99 including VAT and postage (worldwide)

SOFTWARE

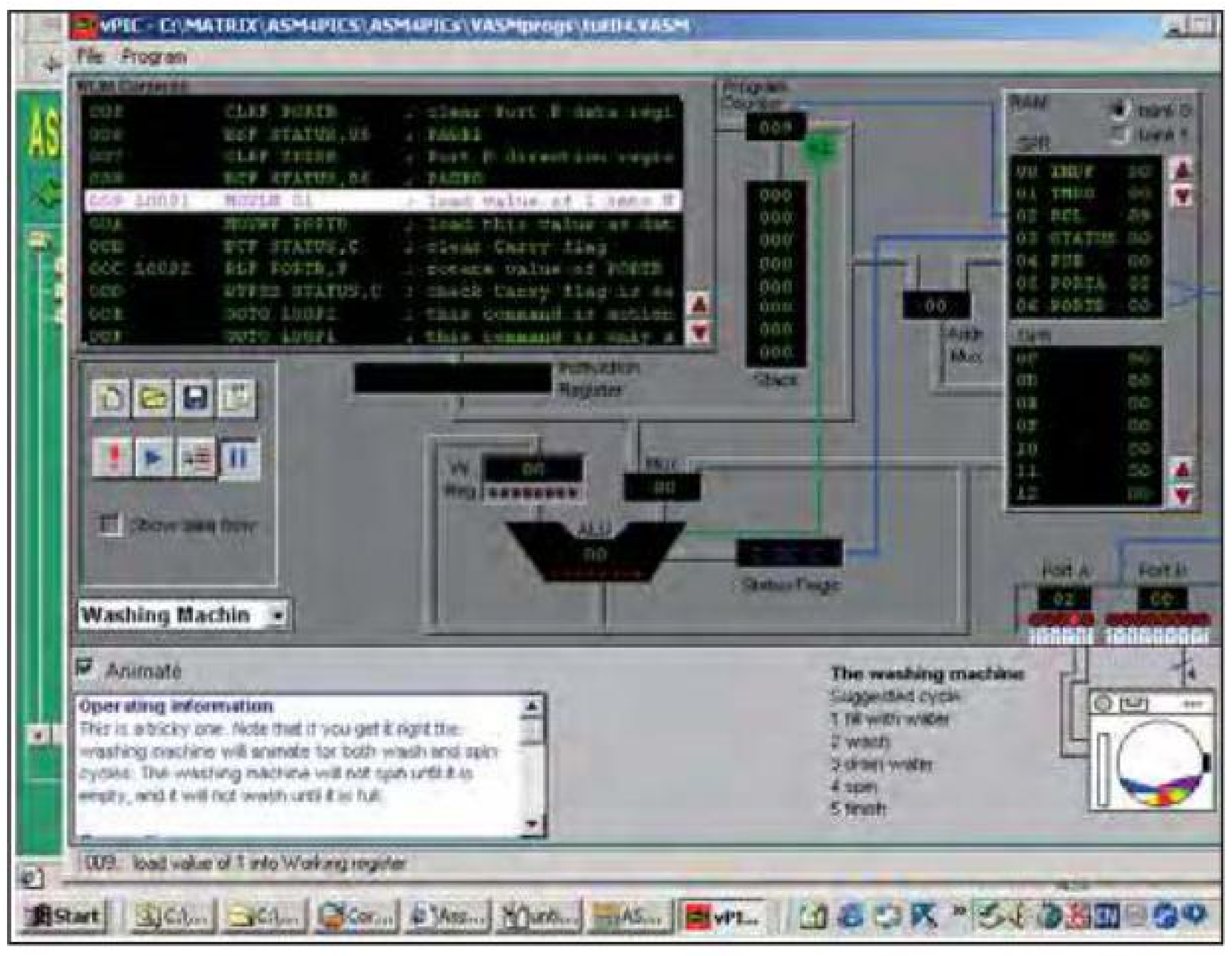
ASSEMBLY FOR PICmicro V7

(Formerly PICtutor)

Assembly for PICmicro microcontrollers V7.0 (previously known as PICtutor) by John Becker contains a complete course in programming the PIC16F84, 16F88 and 16F877a PICmicro microcontroller from Arizona Microchip. It starts with fundamental concepts and extends up to complex programs including watchdog timers, interrupts and sleep modes.

The CD makes use of the latest simulation techniques which provide a superb tool for learning: the Virtual PICmicro microcontroller, this is a simulation tool that allows users to write and execute MPASM assembler code for the PIC16F84 microcontroller on-screen. Using this you can actually see what happens inside the PICmicro MCU as each instruction is executed, which enhances understanding.

- Comprehensive instruction through 45 tutorial sections
- Includes Vlab, a Virtual PICmicro microcontroller: a fully functioning simulator
- Tests, exercises and projects covering a wide range of PICmicro MCU applications
- Includes MPLAB assembler
- Visual representation of a PICmicro showing architecture and functions
- Expert system for code entry helps first time users
- Shows data flow and fetch execute cycle and has challenges (washing machine, lift, crossroads etc.)
- Imports MPASM files.



Operating information
This is a tricky one. Note that if you get it right the washing machine will animate for both wash and spin cycles. The washing machine will not spin until it is empty, and it will not wash until it is full.

The washing machine
Suggested cycle:
1 fill with water
2 wash
3 drain water
4 spin
5 finish

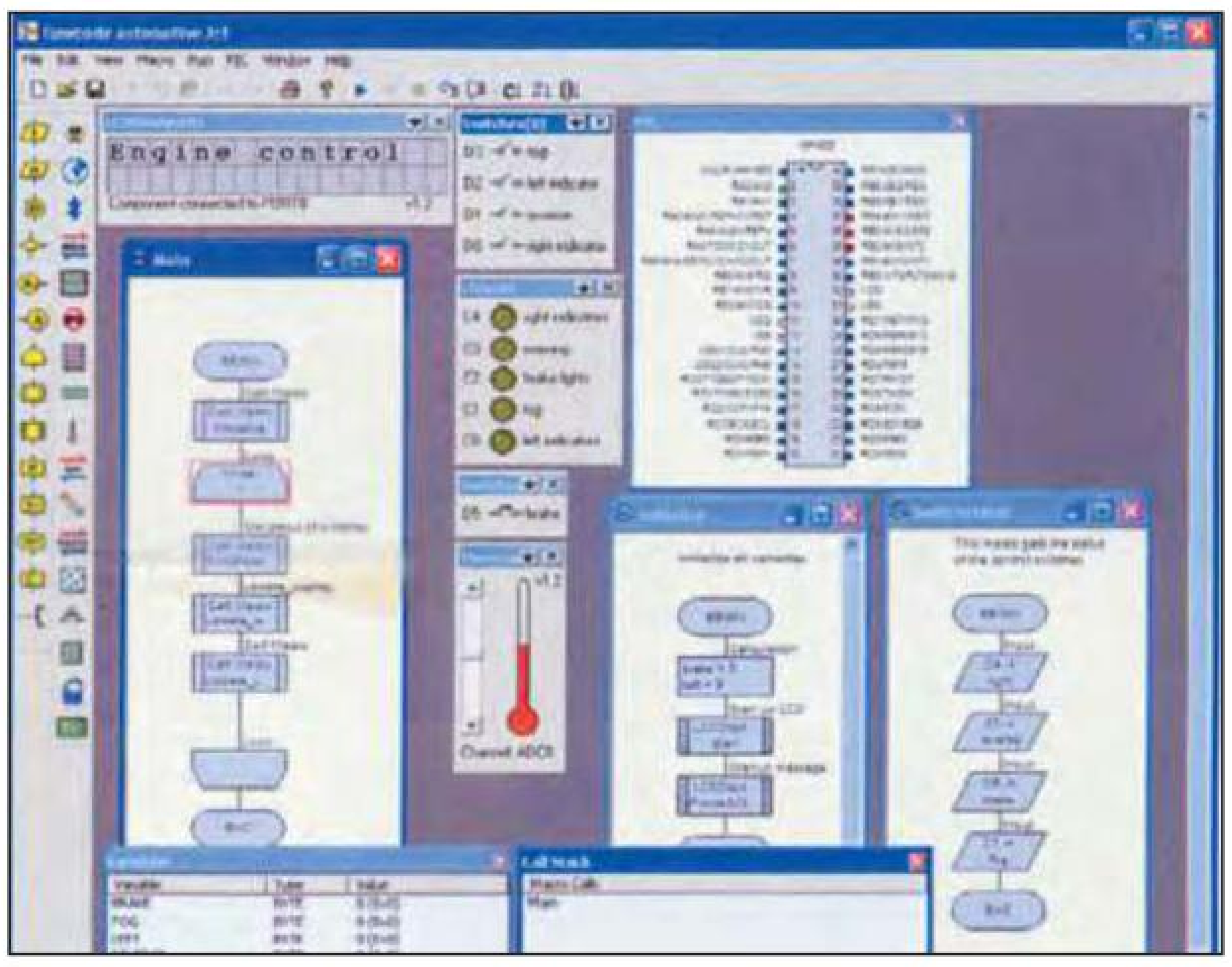
009: load value of 1 into Working register

FLOWCODE FOR PICmicro V8

Flowcode is a very high level language programming system based on flowcharts. Flowcode allows you to design and simulate complex systems in a matter of minutes. A powerful language that uses macros to facilitate the control of devices like 7-segment displays, motor controllers and LCDs. The use of macros allows you to control these devices without getting bogged down in understanding the programming. When used in conjunction with the development board this provides a seamless solution that allows you to program chips in minutes.

- Requires no programming experience
- Allows complex PICmicro applications to be designed quickly
- Uses international standard flow chart symbols
- Full on-screen simulation allows debugging and speeds up the development process.
- Facilitates learning via a full suite of demonstration tutorials
- Produces code for a wide range of devices
- 16-bit arithmetic strings and string manipulation
- Pulse width modulation
- I2C.

Please note: Due to popular demand, Flowcode is now available as a download. Please include your email address and a username (of your choice) on your order. A unique download code will then be emailed to you.



PRICES

Prices for each of the CD-ROMs above are:

(Order form on next page)

(UK and EU customers add VAT to 'plus VAT' prices)

Single License . . . . £99 plus VAT

Site Licence . . . . £499 plus VAT

Flowcode . . . . Contact us for pricing
(choose PIC-8b, PIC-16b, PIC-32b, AVR/Arduino,ARM)

ELECTRONICS TEACH-IN 2

ELECTRONICS TEACH-IN 2 CD-ROM
USING PIC MICROCONTROLLERS A PRACTICAL INTRODUCTION

This *Teach-In* series of articles was originally published in *EPE* in 2008 and, following demand from readers, has been collected together in the *Electronics Teach-In 2* CD-ROM.

The series is aimed at those using PIC microcontrollers for the first time. Each part of the series includes breadboard layouts to aid understanding and a simple programmer project is provided.

Also included are 29 *PIC N' Mix* articles, also republished from *EPE*. These provide a host of practical programming and interfacing information, mainly for those that have already got to grips with using PIC microcontrollers. An extra four part beginners guide to using the C programing language for PIC microcontrollers is also included.

The CD-ROM also contains all of the software for the *Teach-In 2* series and *PIC N' Mix* articles, plus a range of items from Microchip – the manufacturers of the PIC microcontrollers. The material has been compiled by Wimborne Publishing Ltd. with the assistance of Microchip Technology Inc.

CD-ROM **Order code ETI2 CD-ROM** **£9.50**

ELECTRONICS TEACH-IN 3

ELECTRONICS TEACH-IN 3 CD-ROM

The three sections of this CD-ROM cover a very wide range of subjects that will interest everyone involved in electronics, from hobbyists and students to professionals. The first 80-odd pages of Teach-In 3 are dedicated to *Circuit Surgery*, the regular *EPE* clinic dealing with readers' queries on circuit design problems – from voltage regulation to using SPICE circuit simulation software.

The second section – *Practically Speaking* – covers the practical aspects of electronics construction. Again, a whole range of subjects, from soldering to avoiding problems with static electricity and indentifying components, are covered. Finally, our collection of *Ingenuity Unlimited* circuits provides over 40 circuit designs submitted by the readers of *EPE*.

The CD-ROM also contains the complete *Electronics Teach-In 1* book, which provides a broad-based introduction to electronics in PDF form, plus interactive quizzes to test your knowledge, TINA circuit simulation software (a limited version – plus a specially written TINA Tutorial).

The Teach-In 1 series covers everything from Electric Current through to Microprocessors and Microcontrollers and each part includes demonstration circuits to build on breadboards or to simulate on your PC.

CD-ROM **Order code ETI3 CD-ROM** **£8.50**

ELECTRONICS TEACH-IN 4

ELECTRONICS TEACH-IN 4 CD-ROM
A BROAD-BASED INTRODUCTION TO ELECTRONICS.

The Teach-In 4 CD-ROM covers three of the most important electronics units that are currently studied in many schools and colleges. These include, Edexcel BTEC level 2 awards and the electronics units of the Diploma in Engineering, Level 2.

The CD-ROM also contains the full Modern Electronics Manual, worth £29.95. The Manual contains over 800 pages of electronics theory, projects, data, assembly instructions and web links.

A package of exceptional value that will appeal to all those interested in learning about electronics or brushing up on their theory, be they hobbyists, students or professionals.

CD-ROM **Order code ETI4 CD-ROM** **£8.99**

ELECTRONICS TEACH-IN 5

ELECTRONICS TEACH-IN 5
JUMP START

FREE CD-ROM

15 design and build circuit projects dedicated to newcomers or those following courses in school and colleges. The projects are: Moisture Detector, Quiz Machine, Battery Voltage Checker, Solar-Powered Charger, Versatile Theft Alarm, Spooky Circuits, Frost Alarm, Mini Christmas Lights, iPod Speaker, Logic Probe, DC Motor Controller, Egg Timer, Signal Injector Probe, Simple Radio Receiver, Temperature Alarm.

PLUS:

PIC' N MIX – starting out with PIC Microcontrollers and PRACTICALLY SPEAKING – the techniques of project construction.

FREE CD-ROM – The free CD-ROM is the complete Teach-In 2 book providing a practical introduction to PIC Microprocessors plus MikroElektronika, Microchip and L-Tek PoScope software.

160 Pages **Order code ETI5** **£8.99**

ELECTRONICS TEACH-IN BUNDLE – FOR PARTS 3, 4 & 5

CD-ROM **Order code ETIB3 CD-ROM** **£18.95**

ORDERING
ALL PRICES INCLUDE UK POSTAGE

Standard/Student/Basic (Hobbyist) Version price includes postage to most countries in the world EU residents outside the UK add £5 for airmail postage per order

Please send me: **CD-ROM ORDER FORM**

☐ **Assembly for PICmicro V6**
☐ **'C' for 16 Series PICmicro V6**

Version required:
☐ **Single Licence**
☐ **Site licence**

Note: The software on each version is the same, only the licence for use varies.

☐ **PICmicro Multiprogrammer Board and Development Board (hardware)**

☐ **Circuit Wizard – Standard**
☐ **Circuit Wizard – Professional**
☐ **GCSE Electronics**
☐ **TINA Design Suite V11 Basic (Hobbyist)**
☐ **TINA Design Suite V11 (Student)**

☐ **Teach-In 2**
☐ **Teach-In 3**
☐ **Teach-In 4**
☐ **Teach-In Bundle (3, 4 and 5)**

Full name . . . .

Address . . . .

. . . . Post code . . . . Tel . . . .

Email: . . . .

☐ I enclose cheque/PO in £ sterling payable to Practical Electronics for £ . . . .
☐ Please charge my Visa/Mastercard: £ . . . .

Valid From: . . . . Card expiry date: . . . .

Card No: . . . .

Card Security Code . . . .
(The last 3 digits on or just under the signature strip)

Single License and Site License Versions – overseas readers add £5 to the basic price of each order for airmail postage (**do not add VAT** unless you live in an EU (European Union) country, then add VAT at 20% or provide your official VAT registration number).

Send your order to:
Electron Publishing Limited
113 Lynwood Drive
Wimborne
Dorset BH21 1UU

To order by phone call: **01202 880299**

Goods are normally sent within seven days

E-mail: shop@electronpublishing.com

Online shop:

www.epemag.com

DIRECT BOOK SERVICE

Teach-In 2017

Introducing the BBC micro:bit

For Beginners of Any Age
Python Coding on the BBC micro:bit
Jim Gatenby

PYTHON CODING ON THE BBC MICRO:BIT

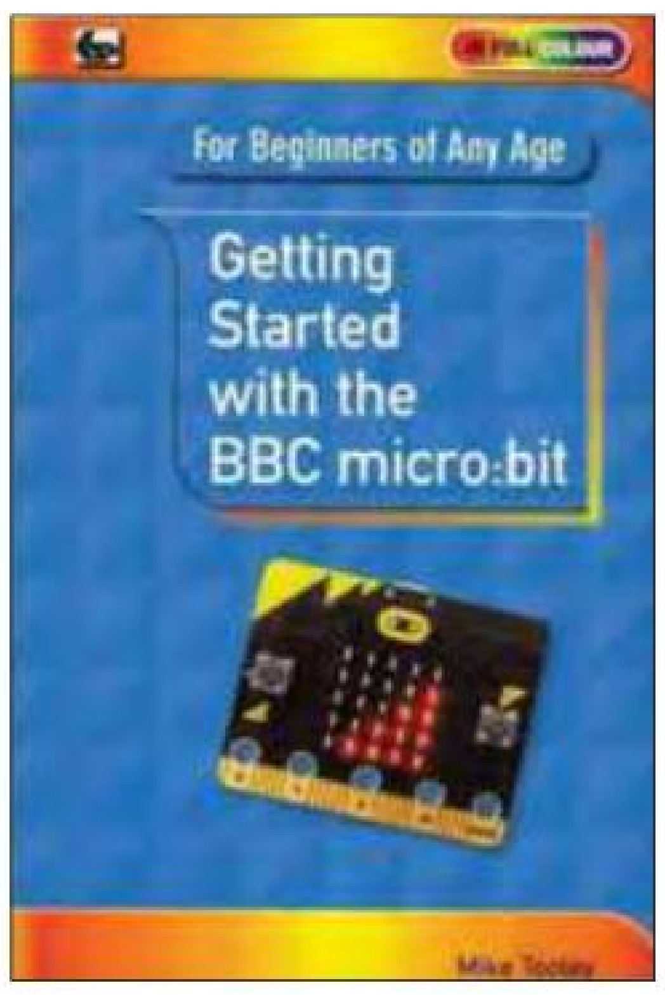
Jim Gatenby

Python is the leading programming language, easy to learn and widely used by professional programmers. This book uses MicroPython, a version of Python adapted for the BBC Micro:bit.

Among the many topics covered are: main features of the BBC micro:bit including a simulation in a web browser screen; various levels of programming languages; Mu Editor for writing, saving and retrieving programs, with sample programs and practice exercises; REPL, an interactive program for quickly testing lines of code; scrolling messages, creating and animating images on the micro:bit's LEDs; playing and creating music, sounds and synthesized speech; using the on-board accelerometer to detect movement of the micro:bit on three axes; glossary of computing terms.

This book is written using plain English, avoids technical jargon wherever possible and covers many of the coding instructions and methods which are common to most programming languages. It should be helpful to beginners of any age, whether planning a career in computing or writing code as an enjoyable hobby.

118 Pages **Order code PYTH MBIT** £7.99



GETTING STARTED WITH THE BBC MICRO:BIT

Mike Tooley

Not just an educational resource for teaching youngsters coding, the BBC micro:bit is a tiny low cost, low-profile ARM-based single-board computer. The board measures 43mm × 52mm but despite its diminutive footprint it has all the features of a fully fledged microcontroller together with a simple LED matrix display, two buttons, an accelerometer and a magnetometer.

Mike Tooley's book will show you how the micro:bit can be used in a wide range of applications from simple domestic gadgets to more complex control systems such as those used for lighting, central heating and security applications. Using Microsoft Code Blocks, the book provides a progressive introduction to coding as well as interfacing with sensors and transducers.

Each chapter concludes with a simple practical project that puts into practice what the reader has learned. The featured projects include an electronic direction finder, frost alarm, reaction tester, battery checker, thermostatic controller and a passive infrared (PIR) security alarm.

No previous coding experience is assumed, making this book ideal for complete beginners as well as those with some previous knowledge. Self-test questions are provided at the end of each chapter, together with answers at the end of the book. So whatever your starting point, this book will take you further along the road to developing and coding your own real-world applications.

108 Pages **Order code BBC MBIT** £7.99

THEORY AND REFERENCE

PRACTICAL ELECTRONICS HANDBOOK – 6th Ed
Ian Sinclair
440 pages **Order code NE21** £33.99

STARTING ELECTRONICS – 4th Ed
Keith Brindley
296 pages **Order code ELSEV100** £18.99

ELECTRONIC CIRCUITS – FUNDAMENTALS & APPLICATIONS – Updated version
Mike Tooley
400 pages **Order code TF43** £32.99

FUNDAMENTAL ELECTRICAL AND ELECTRONIC PRINCIPLES – 3rd Ed
C.R. Robertson
368 pages **Order code TF47** £21.99

A BEGINNER'S GUIDE TO TTL DIGITAL ICs
Robert Penfold
142 pages **OUT OF PRINT BP332** £5.45

UNDERSTANDING ELECTRONIC CONTROL SYSTEMS
Owen Bishop
228 pages **Order code NE35** £36.99

The books listed here have been selected by the *Practical Electronics* editorial staff as being of special interest to everyone involved in electronics and computing. They are supplied by mail order direct to your door.

FOR A FULL DESCRIPTION OF THESE BOOKS AND CD-ROMS SEE THE SHOP ON OUR WEBSITE

www.epemag.com

All prices include UK postage

MICROPROCESSORS

INTERFACING PIC MICROCONTROLLERS – 2nd Ed
Martin Bates
298 pages **Order code NE48** £34.99

PROGRAMMING 16-BIT PIC MICROCONTROLLERS IN C – LEARNING TO FLY THE PIC24
Lucio Di Jasio (Application Segments Manager, Microchip, USA)
496 pages + CD-ROM **Order code NE45** £38.00

INTRODUCTION TO MICROPROCESSORS AND MICROCONTROLLERS – 2nd Ed
John Crisp
222 pages **Order code NE31** £29.99

THE PIC MICROCONTROLLER YOUR PERSONAL INTRODUCTORY COURSE – 3rd Ed
John Morton
270 pages **Order code NE36** £25.00

PIC IN PRACTICE – 2nd Ed
David W. Smith
308 pages **Order code NE39** £24.99

MICROCONTROLLER COOKBOOK
Mike James
240 pages **Order code NE26** £36.99

BOOK ORDERING DETAILS

All prices include UK postage.

For postage, add £3 per book to Europe, £4 for rest of the world per book.

CD-ROM prices include VAT and/or postage to anywhere in the world.

Send a cheque, (£ sterling only) made payable to: **Practical Electronics** or credit card details (Visa or Mastercard) to:

Electron Publishing Limited,
113 Lynwood Drive, Wimborne, Dorset BH21 1UU

Books are normally sent within seven days of receipt of order.

Please check price (see latest issue of Practical Electronics or website) before ordering from old lists.

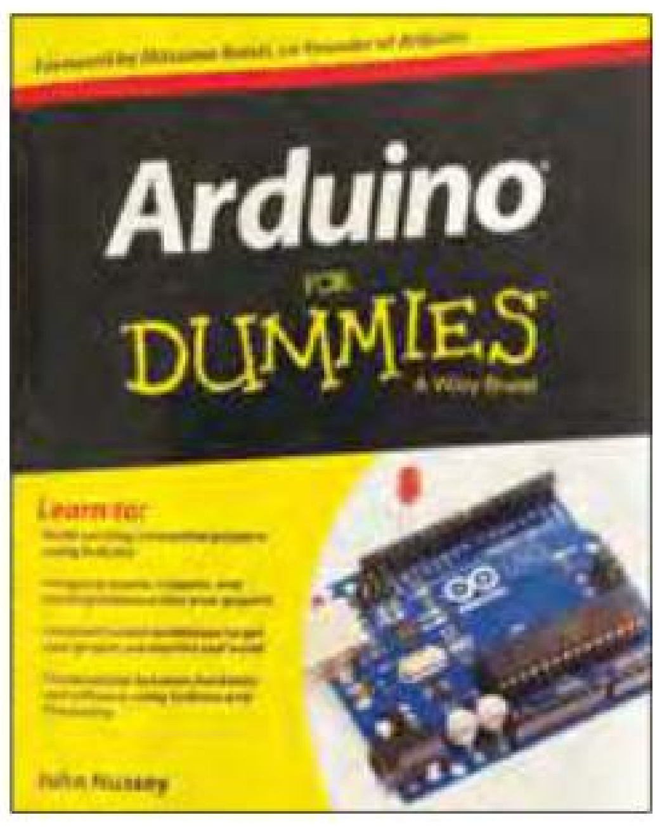
For a full description of these books please see the shop on our website.
Tel: 01202 880299 – Email: shop@electronpublishing.com

Order from our online shop at: www.epemag.com

ARDUINO

Teach-In 2016

See opposite for our popular introduction to the Arduino



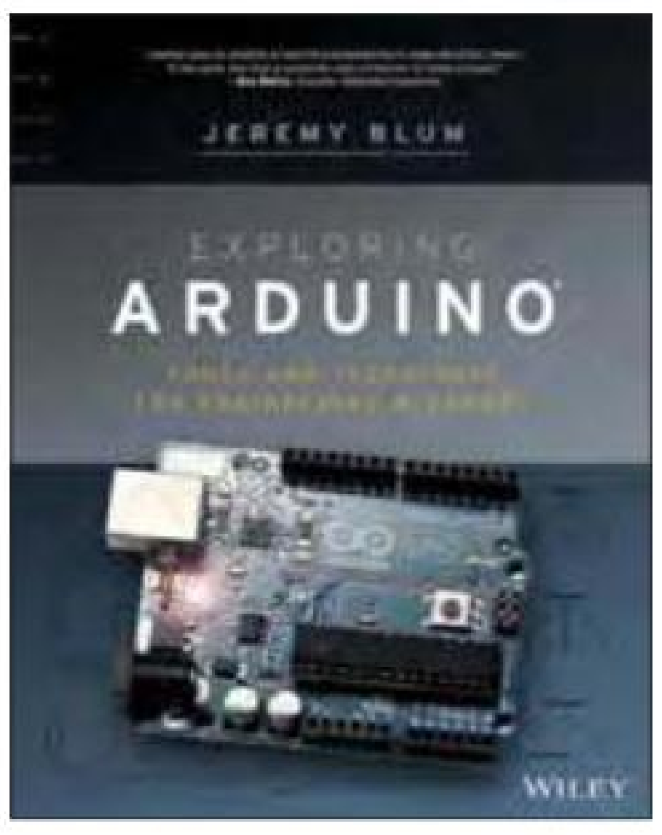
ARDUINO FOR DUMMIES

John Nussey

Arduino is no ordinary circuit board. Whether you're an artist, a designer, a programmer, or a hobbyist, Arduino lets you learn about and play with electronics. You'll discover how to build a variety of circuits that can sense or control real-world objects, prototype your own product, and even create interactive artwork. This handy guide is exactly what you need to build your own Arduino project – what you make is up to you!

- **Learn by doing** – start building circuits and programming your Arduino with a few easy examples – right away!
- **Easy does it** – work through Arduino sketches line by line, and learn how they work and how to write your own.
- **Solder on!** – don't know a soldering iron from a curling iron? No problem! You'll learn the basics and be prototyping in no time.
- **Kitted out** – discover new and interesting hardware to turn your Arduino into anything from a mobile phone to a Geiger counter.
- **Become an Arduino savant** – find out about functions, arrays, libraries, shields and other tools that let you take your Arduino project to the next level
- **Get social** – teach your Arduino to communicate with software running on a computer to link the physical world with the virtual world

438 Pages **Order code ARDDUM01** £19.99



EXPLORING ARDUINO

Jeremy Blum

Arduino can take you anywhere. This book is the roadmap.

Exploring Arduino shows how to use the world's most popular microcontroller to create cool, practical, artistic and educational projects. Through lessons in electrical engineering, programming and human-computer interaction, this book walks you through specific, increasingly complex projects, all the while providing best practices that you can apply to your own projects once you've mastered these. You'll acquire valuable skills – and have a whole lot of fun.

- **Explore** the features of commonly used Arduino boards
- **Use** Arduino to control simple tasks or complex electronics
- **Learn** principles of system design, programming and electrical engineering
- **Discover** code snippets, best practices and system schematics you can apply to your original projects
- **Master** skills you can use for engineering endeavours in other fields and with different platforms

357 Pages **Order code EXPARD01** £26.99

COMPUTING AND ROBOTICS

NEWNES INTERFACING COMPANION
Tony Fischer-Cripps
295 pages **Order code NE38** £41.00

HOW TO BUILD A COMPUTER MADE EASY
Robert Penfold
120 pages **Order code BP707** £8.49

EASY PC CASE MODDING
Robert Penfold
192 pages + CD-ROM **Order code BP542** £8.99

FREE DOWNLOADS TO PEP-UP AND PROTECT YOUR PC
Robert Penfold
128 pages **Order code BP722** £7.99

WINDOWS XP EXPLAINED
N. Kantaris and P.R.M. Oliver
264 pages **Order code BP514** £7.99

THE INTERNET – TWEAKS, TIPS AND TRICKS
Robert Penfold
128 pages **Order code BP721** £7.99

eBAY – TWEAKS, TIPS AND TRICKS
Robert Penfold
128 pages **Order code BP716** £7.50

AN INTRODUCTION TO eBAY FOR THE OLDER GENERATION
Cherry Nixon
120 pages **Order code BP709** £8.49

HOW TO FIX YOUR PC PROBLEMS
Robert Penfold
128 pages **Order code BP705** £8.49

AN INTRODUCTION TO WINDOWS VISTA
P.R.M. Oliver and N. Kantarris
120 pages **Order code BP703** £8.49

WINDOWS 8.1 EXPLAINED
180 Pages **Order code BP747** £10.99

AN INTRODUCTION TO THE NEXUS 7
118 Pages **Order code BP744** £8.99

COMPUTING FOR THE OLDER GENERATION
Jim Gatenby
308 pages **Order code BP601** £8.99

ANDROIDS, ROBOTS AND ANIMATRONS
Second Edition – John Iovine
224 pages **Order code MGH1** £16.99

ROBOT BUILDERS COOKBOOK
Owen Bishop
366 pages **Order code NE46** £26.00

INTRODUCING ROBOTICS WITH LEGO MINDSTORMS
Robert Penfold
288 pages + **Order code BP901** £14.99

MORE ADVANCED ROBOTICS WITH LEGO MINDSTORMS
Robert Penfold
298 pages **Order code BP902** £14.99

HOW TO FIX YOUR PC PROBLEMS
Robert Penfold
128 pages **Order code BP705** £8.49

WINDOWS 7 – TWEAKS, TIPS AND TRICKS
Andrew Edney
120 pages **Order code BP708** £8.49

GETTING STARTED IN COMPUTING FOR THE OLDER GENERATION
Jim Gatenby
120 pages **Order code BP704** £8.49

WINDOWS 8.1 EXPLAINED
Noel Kantaris
180 Pages **Order code BP747** £10.99

COMPUTING WITH A LAPTOP FOR THE OLDER GENERATION
Robert Penfold
120 pages **Order code BP702** £8.49

AN INTRODUCTION TO EXCEL SPREADSHEETS
Jim Gatenby
18 pages **Order code BP701** £8.49

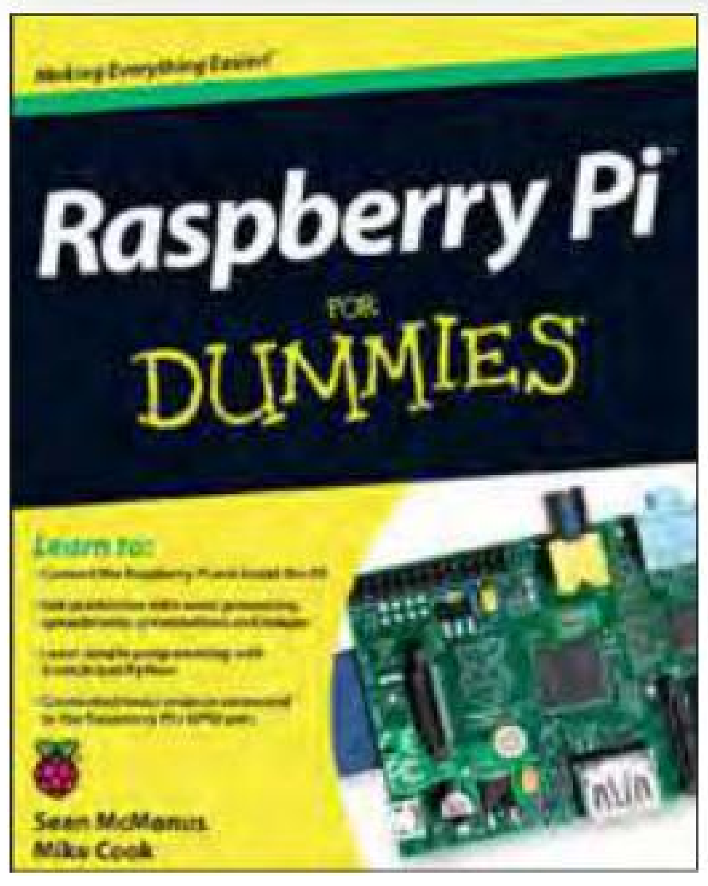
KINDLE FIRE HDX EXPLAINED
118 Pages **Order code BP743** £8.99

AUDIO & VIDEO

VALVE AMPLIFIERS – 4th Ed
Morgan Jones
288 pages **Order code ELSEV33** £46.99

BUILDING VALVE AMPLIFIERS
Morgan Jones
368 pages **Order code NE40** £29.00

RASPBERRY PI



RASPBERRY Pi FOR DUMMIES

Sean McManus and Mike Cook

Write games, compose and play music, even explore electronics – it's easy as Pi! The Raspberry Pi offers a plateful of opportunities, and this great resource guides you step-by-step, from downloading, copying, and installing the software to learning about Linux and finding cool new programs for work, photo editing, and music. You'll discover how to write your own Raspberry Pi programs, create fun games, and much more!

Open this book and find:
What you can do with Python; Ways to use the Raspberry Pi as a productivity tool; How to surf the web and manage files; Secrets of Sonic Pi music programming; A guide to creating animations and arcade games; Fun electronic games you can build; How to build a 3D maze in Minecraft; How to play music and videos on your Raspberry Pi.

400 Pages **Order code RPiDUM01** £17.99

RASPBERRY Pi MANUAL: A practical guide to the revolutionary small computer
176 pages **Order code H001** £17.99

RASPBERRY Pi USER-GUIDE – 4th Ed
262 pages **Order code JW001** £20.90

PROGRAMMING THE RASPBERRY Pi
192 pages **Order code MGH4** £10.99

GETTING STARTED WITH RASPBERRY Pi
164 pages **Order code OR01** £11.50

TEACH-IN BOOKS

ELECTRONICS TEACH-IN 6

ELECTRONICS TEACH-IN 6
FREE DVD-ROM
£8.99
RASPBERRY Pi
A COMPREHENSIVE GUIDE TO RASPBERRY Pi
PLUS

ONLY AVAILABLE ON CD-ROM

ELECTRONICS TEACH-IN 6
A COMPREHENSIVE GUIDE TO RASPBERRY Pi

Mike & Richard Tooley

Teach-In 6 contains an exciting series of articles that provides a complete introduction to the Raspberry Pi, the low-cost computer that has taken the education and computing world by storm.

This latest book in our Teach-In series will appeal to electronic enthusiasts and computer buffs wanting to get to grips with the Raspberry Pi.

Anyone considering what to do with their Pi, or maybe they have an idea for a project but don't know how to turn it into reality, will find Teach-In 6 invaluable. It covers: Programming, Hardware, Communications, Pi Projects, Pi Class, Python Quickstart, Pi World, and Home Baking.

The CD-ROM also contains all the necessary software for the series so that readers can get started quickly and easily with the projects and ideas covered.

160 Pages **Order code ETI6** **£8.99**

ELECTRONICS TEACH-IN 7 (Includes free CD-ROM)

ELECTRONICS TEACH-IN 7
FREE CD-ROM
£8.99
DISCRETE LINEAR CIRCUIT DESIGN
PLUS...
AUDIO OUT
PRACTICALLY SPEAKING

RUNNING LOW – ORDER NOW!

ELECTRONICS TEACH-IN 7
DISCRETE LINEAR CIRCUIT DESIGN

Mike & Richard Tooley

Teach-In 7 is a complete introduction to the design of analogue electronic circuits. Ideal for everyone interested in electronics as a hobby and for those studying technology at schools and colleges. Supplied with a free cover-mounted CD-ROM containing all the circuit software for the course, plus demo CAD software for use with the Teach-In series

Discrete Linear Circuit Design* Understand linear circuit design* Learn with 'TINA' – modern CAD software* Design simple, but elegant circuits* Five projects to build: Pre-amp, Headphone Amp, Tone Control, VU-meter, High Performance Audio Power Amp.

PLUS

Audio Out – an analogue expert's take on specialist circuits
Practically Speaking – the techniques of project building

160 Pages **Order code ETI7** **£8.99**

ELECTRONICS TEACH-IN 8

ELECTRONICS TEACH-IN 8
FREE CD-ROM
£8.99
INTRODUCING THE ARDUINO
PLUS...
PIC n' MIX

ONLY AVAILABLE ON CD-ROM

ELECTRONICS TEACH-IN 8
INTRODUCING THE ARDUINO

Mike & Richard Tooley

Hardware – learn about components and circuits; Programming – powerful integrated development system; Microcontrollers – understand control operations; Communications – connect to PCs and other Arduinos

This exciting series has been designed for electronics enthusiasts who want to get to grips with the inexpensive, immensely popular Arduino microcontroller, as well as coding enthusiasts who want to explore hardware and interfacing. Teach-In 8 will provide a one-stop source of ideas and practical information.

The Arduino offers a remarkably effective platform for developing a huge variety of projects; from operating a set of Christmas tree lights to remotely controlling a robotic vehicle through wireless or the Internet. Teach-In 8 is based around a series of practical projects with plenty of information to customise each project.

This book also includes *PIC n' Mix*: PICs and the PICkit 3 - A Beginners guide by Mike O'Keefe and *Circuit Surgery* by Ian Bell - State Machines part 1 and 2.

The CD-ROM includes files for Teach-In 8 plus Microchip MPLAB IDE XC8 8-bit Compiler and PICkit 3 User Guide. Also included is Lab-Nation Smartscope software.

160 Pages **Order code ETI8** **£8.99**

VISIT OUR WEBSITE FOR MORE BOOKS AND FAST, EASY ONLINE ORDERING
WWW.EPEMAG.COM

BOOK ORDER FORM

Full name: ....

Address: ....

....

.... Post code: .... Telephone No: ....

Email: ....

☐ I enclose cheque/PO payable to Practical Electronics for £ ....

☐ Please charge my card £ .... Card expiry date...

Card Number ....

Valid From Date .... Card Security Code .... (The last three digits on or just below the signature strip)

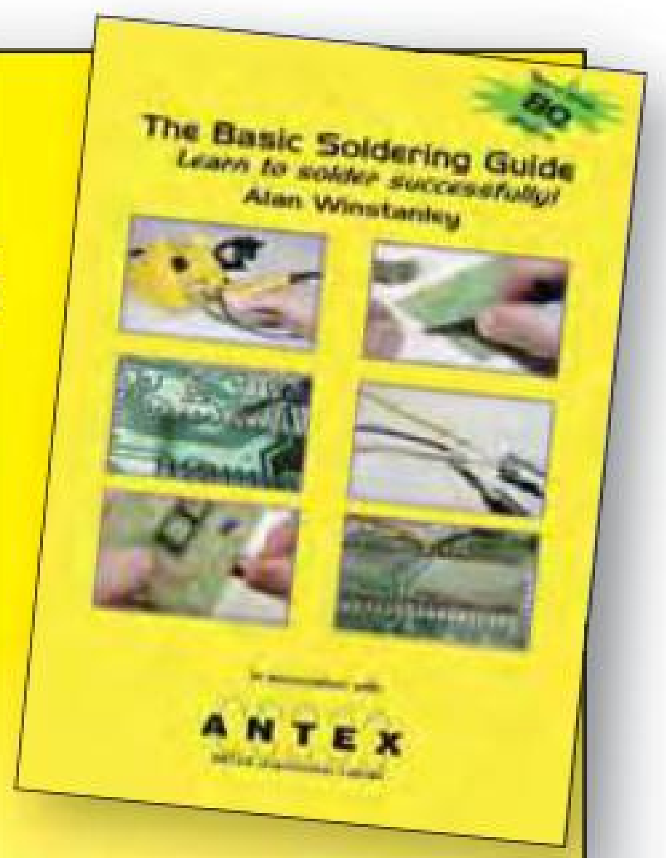
Please send book order codes: ....

....

....

Please continue on separate sheet of paper if necessary

THE BASIC SOLDERING GUIDE



LEARN TO SOLDER SUCCESSFULLY!
ALAN WINSTANLEY

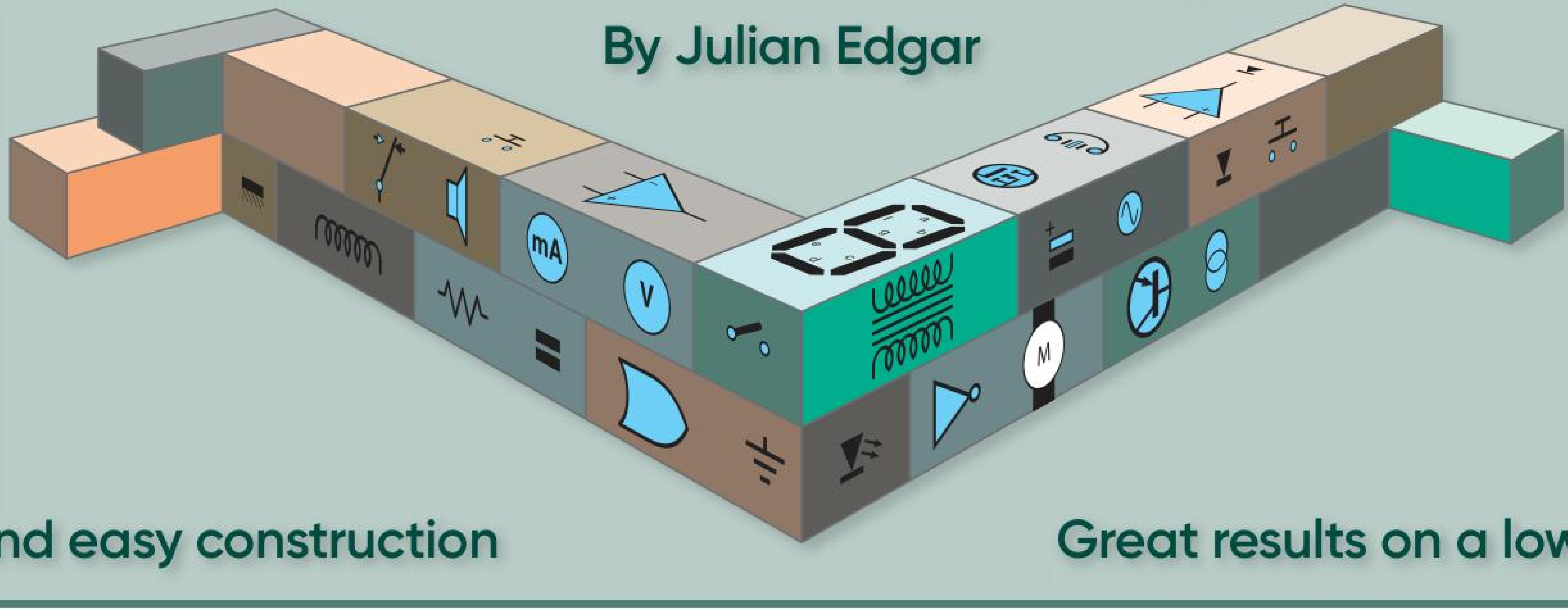
The No.1 resource for learning all the basic aspects of electronics soldering by hand.

With more than 80 high quality colour photographs, this book explains the correct choice of soldering irons, solder, fluxes and tools. The techniques of how to solder and desolder electronic components are then explained in a clear, friendly and non-technical fashion so you'll be soldering successfully in next to no time! The book also includes sections on reflow soldering and desoldering techniques, potential hazards, useful resources and a very useful troubleshooting guide.

Also ideal for those approaching electronics from other industries, the *Basic Soldering Guide Handbook* is the best resource of its type, and thanks to its excellent colour photography and crystal clear text, the art of soldering can now be learned by everyone!

86 Pages **Order code AW1** **£9.99**

Electronic Building Blocks



By Julian Edgar

Quick and easy construction

Great results on a low budget

High-powered voice alarm

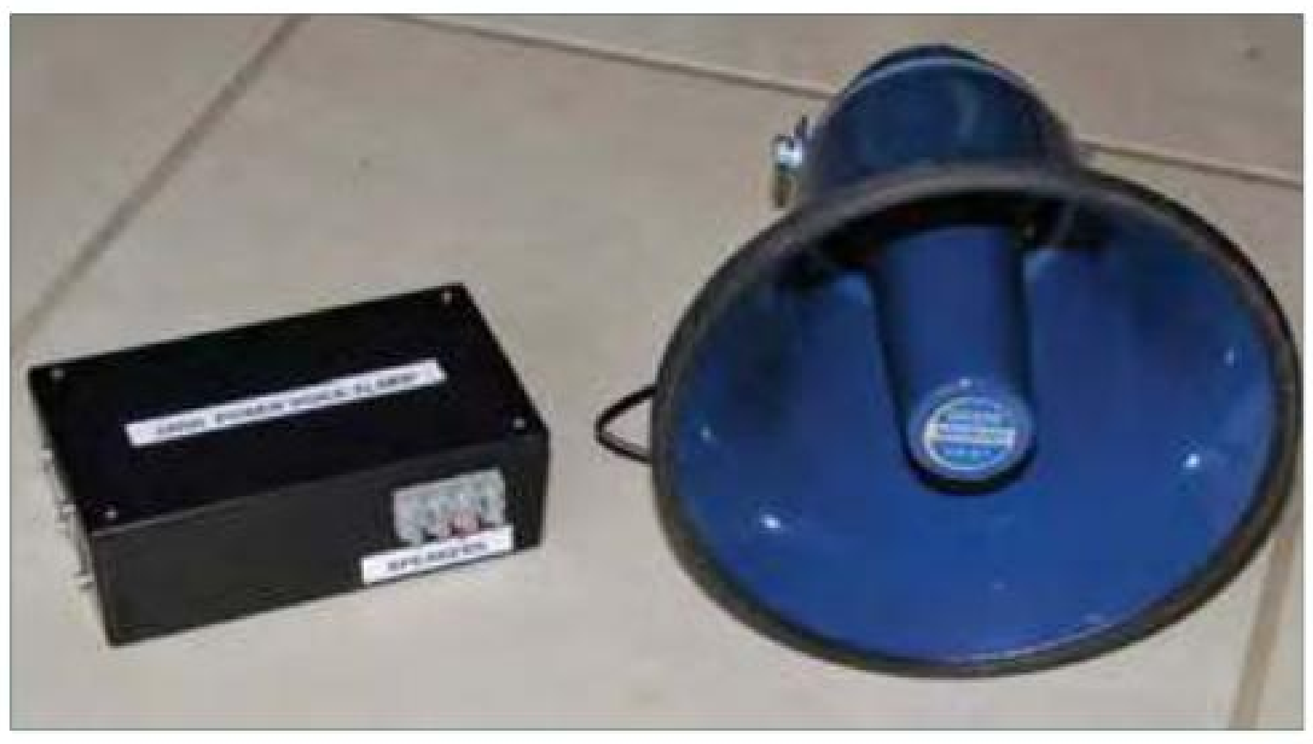


Fig.1. The high-powered voice alarm is built into a plastic box. It is pictured with a single 15W horn speaker but can drive two, creating a very loud voice warning. You can easily add the sounds of a siren to the recorded message, making it especially effective.

Do you want an alarm that literally shouts voice messages when triggered? That's just what this project will do. It's ideal as a fire alarm (my application) or burglar alarm. You can record any message you like, complete with background sirens. No one could sleep through it, and it would take a very brave burglar to ignore it – especially if the sound is projected through externally mounted horn speakers.

Such a system would have once been the stuff of science fiction, but now it's cheap and easy. Take a look at this parts list:

- ISD1820 Sound Voice Recording Playback Module
- LM317 DC-DC 1.5A Step Down Power Supply Module
- TDA7297 Amplifier Board
- 15W 8Ω horn speakers ×2

These items are available from many eBay sellers – simply search using the above terms. Then just add a box, some low-cost hardware and wiring.

So how do the different components work – and how does the system come together? Let's take a look.

Voice module

The sounds played back through the speakers are created by the voice module. To find this pre-built board, do an eBay search under 'ISD1820 module' (at the time of writing, eBay item 223868846287, £2.38 incl p&p). These modules are equipped with:

- ISD1820 sound record/playback IC
- Three pushbuttons
- Configurable links
- Power connections via header pins (note: 3-5V – see below for the power supply)
- Connector for an external speaker (instead, we will feed this output into the amplifier module)
- On-board microphone.

The recorded message can be up to 10 seconds long and the quality of the recording is quite good.

After you have applied power to the module, record a message by pressing the 'REC' button and speaking into the microphone. Best quality is achieved with your mouth close to the microphone. The message is retained even if power is removed. Linking the on-board 'P-E' pins causes the message to be played in a continuous loop whenever power is again connected to the board – just what we want in this 'alarm' application.

If you want the sound of sirens, the easiest route is to find an appropriate YouTube video and play that back through your PC speakers as you are recording. For example, you can start the message with a fire siren, pause the YouTube playback as you add a voice warning – 'Fire in the kitchen! Fire in the kitchen. Evacuate! Evacuate! – and then finish the message with the sound of the siren. Played on a continuous loop, this

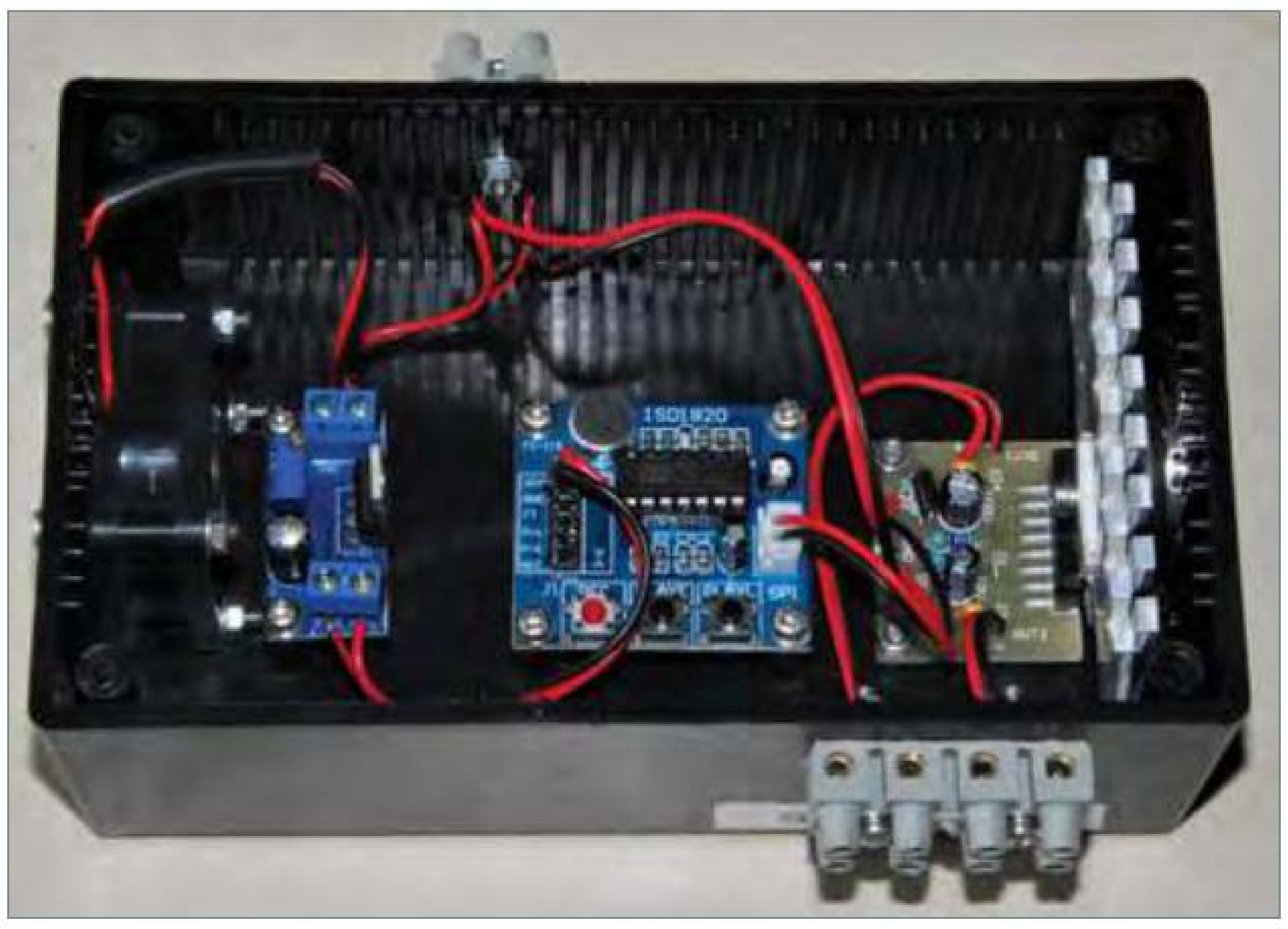


Fig.2. Inside the box, left to right: cooling fan, voice module power supply, voice module, amplifier module and heatsink. The front terminal block is for the two speakers, rear block is for the 12V power connection.

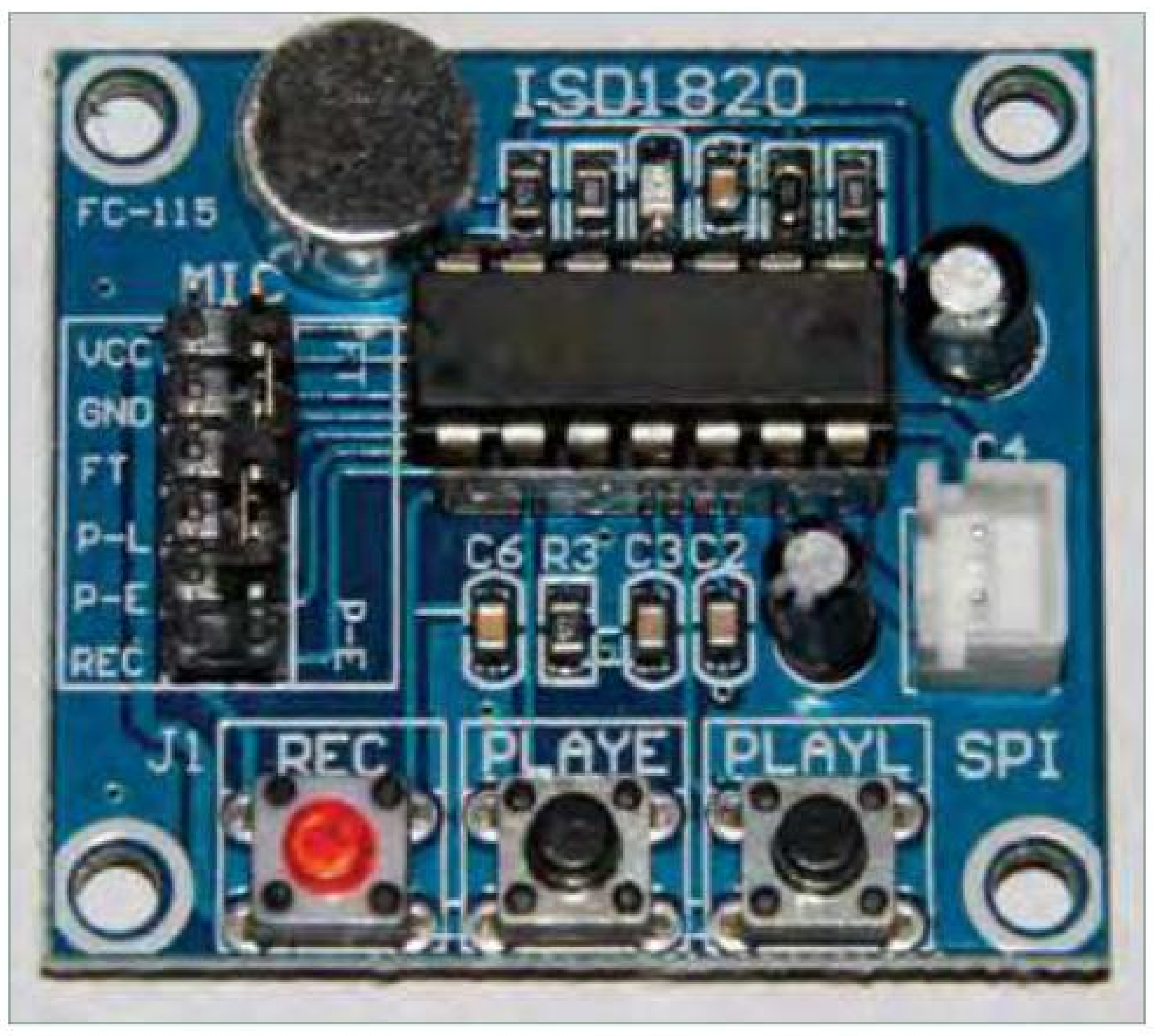


Fig.3. The required message is recorded on this ISD1820 module. It can be configured to play the message on a continuous loop when power is applied – just what we want in this application.

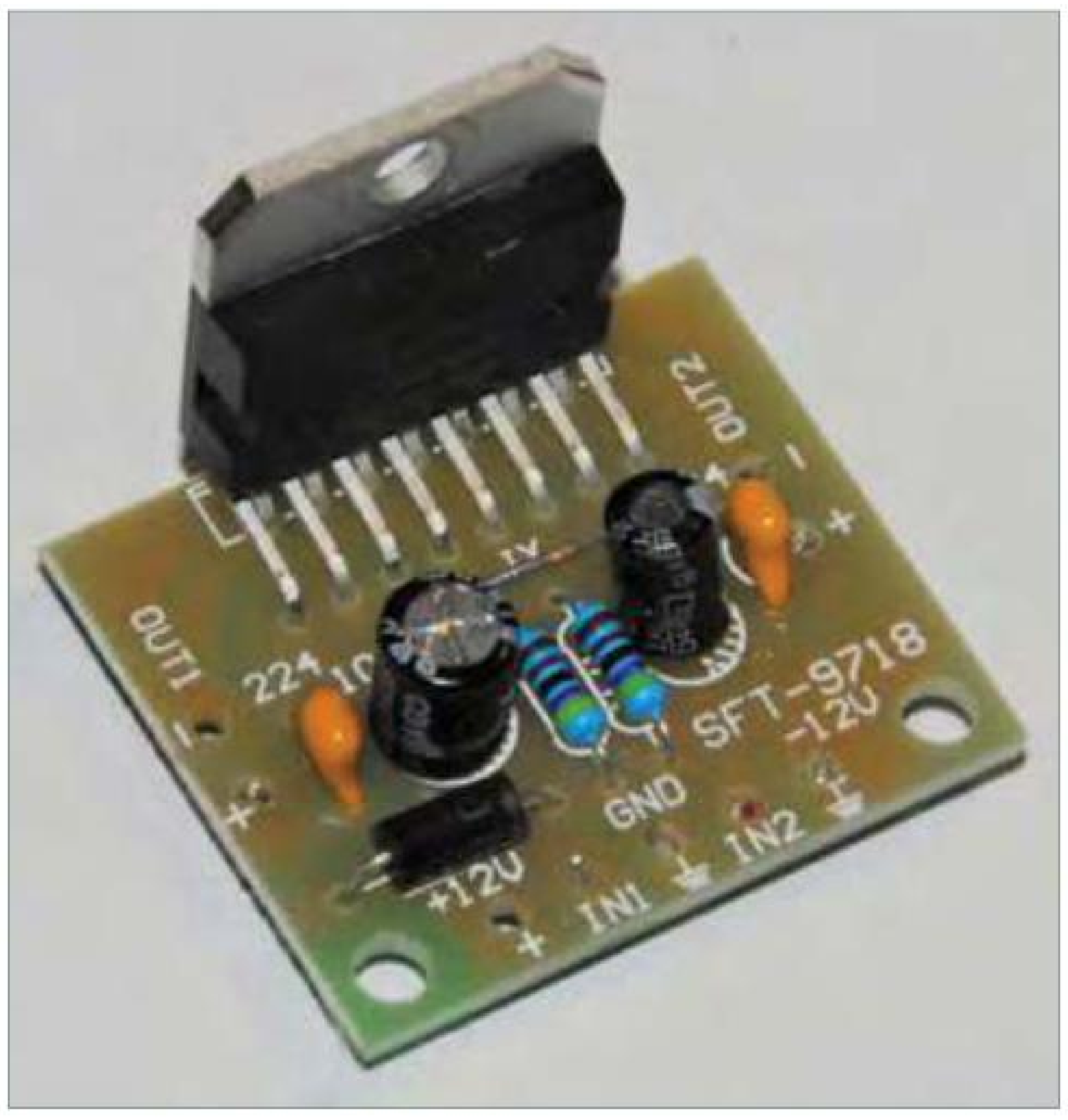


Fig.4. Audio power is provided by a TDA7297 amplifier board. It is available in kit or prebuilt forms. Maximum power is 15W ×2 – a lot in this application!

is very effective. And, if you're not happy with the message, just record it again.

Voice module power supply

To provide the 3.3-5V required by the voice module, we use a small power supply. (The rest of the system works off a nominal 12V.) This lower voltage is easy and cheap to provide with an LM317 adjustable linear regulator board; for example, eBay item 254489419130 (at the time of writing £1.86 incl p&p). Most of these pre-built eBay modules come with a heatsink, but in our application the current draw of the voice module is so small that no heatsink is needed.

TDA7297 Amplifier

Now let's look at what makes this system really audible – the amplifier module. Based on the TDA7297 module, these amplifier chips can output 15W ×2 – in this application, that's plenty! The one I bought is a simple kit – just eight components and takes literally only a few minutes to assemble; for example, eBay item 153801136438 (at the time of writing £1.61 incl p&p). The board is well labelled: (+) and (–) 12V power, signal inputs and two speaker outputs. The kit has no volume control – if you want a volume control, select one of the many other TDA7297 boards that come with this feature.

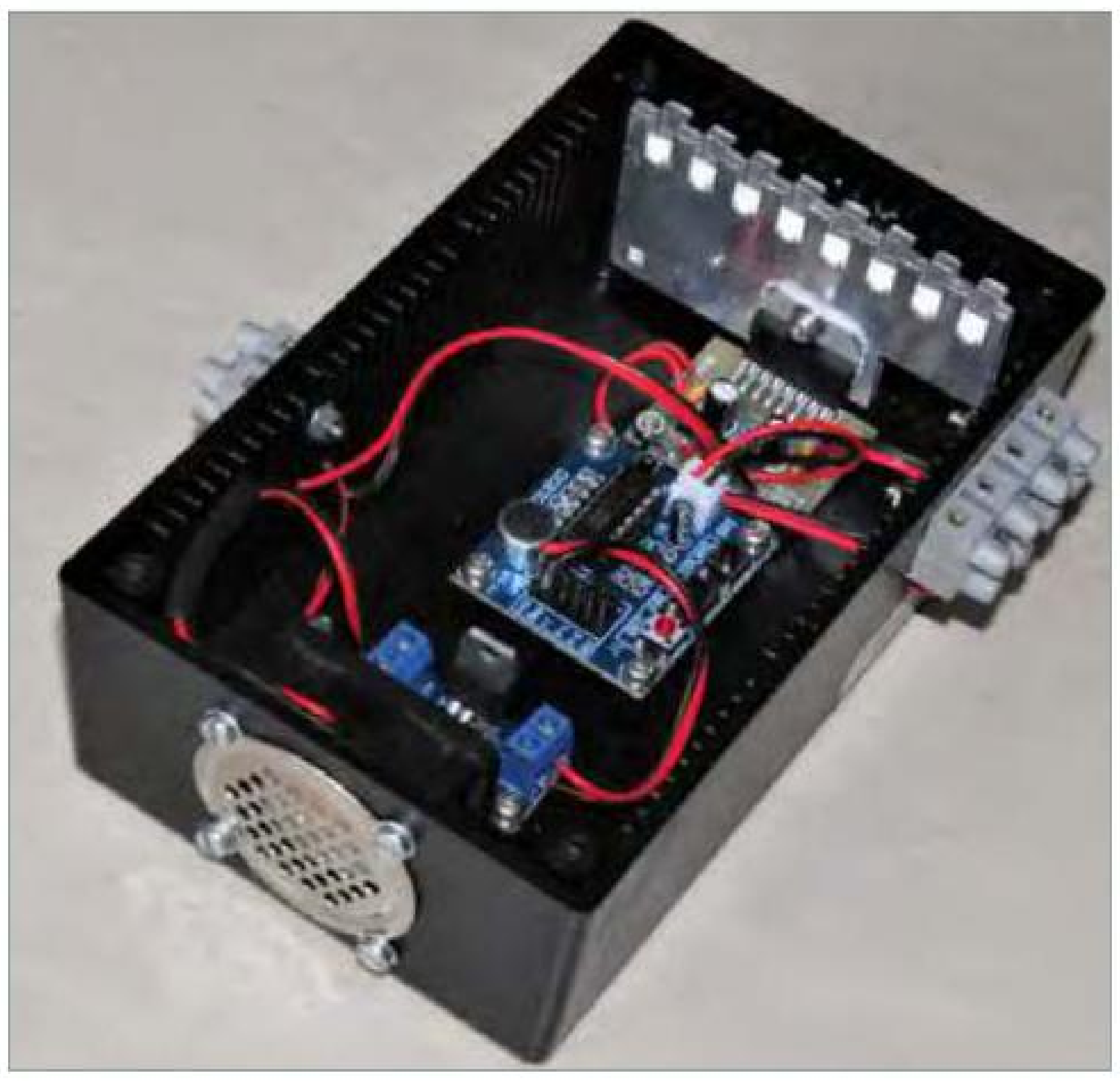


Fig.6. End view of the enclosure showing the grille over the fan. A similar grille is located at the other end behind the heatsink, allowing airflow through the enclosure. (These grilles are sold for use in cupboard doors.)

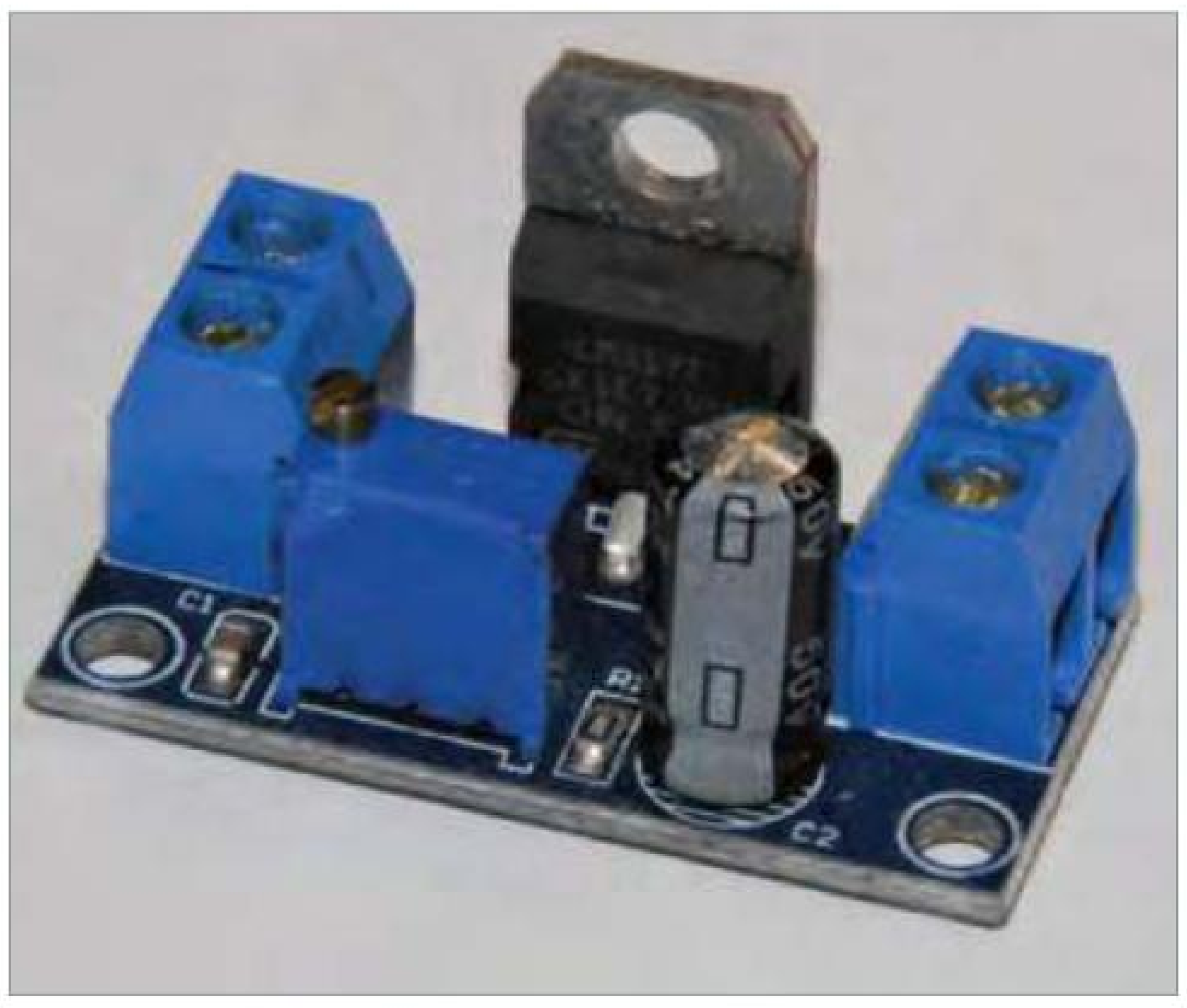


Fig.5. The voice module requires 3-5V, which is provided by this adjustable LM317 power supply module.

Note that a substantial heatsink (or fan cooling) is needed when the module is being run at full power – as is the case in this project.

Building the project

As with many electronic projects, it's best to first set the system up in 'quick and dirty' form on the bench and see if it all works.

Start off by feeding 12V to the LM317 module and adjust its output to about 3.5V, as measured with a multimeter. (Be careful – these boards are not usually reverse polarity or short-circuit proof!)

Connect the power supply's output to the power terminals on the ISD1820 module. (You will need to solder to the on-board pins.) Temporarily connect a speaker to the output leads of the voice module – these are the flying leads provided with a plug to connect to the board.

Apply power and press the REC button. Speak into the microphone and record a message. Disconnect power, link the 'P-E' pins with the provided link and re-apply power. Your message should now play back through the speaker on a continuous loop.

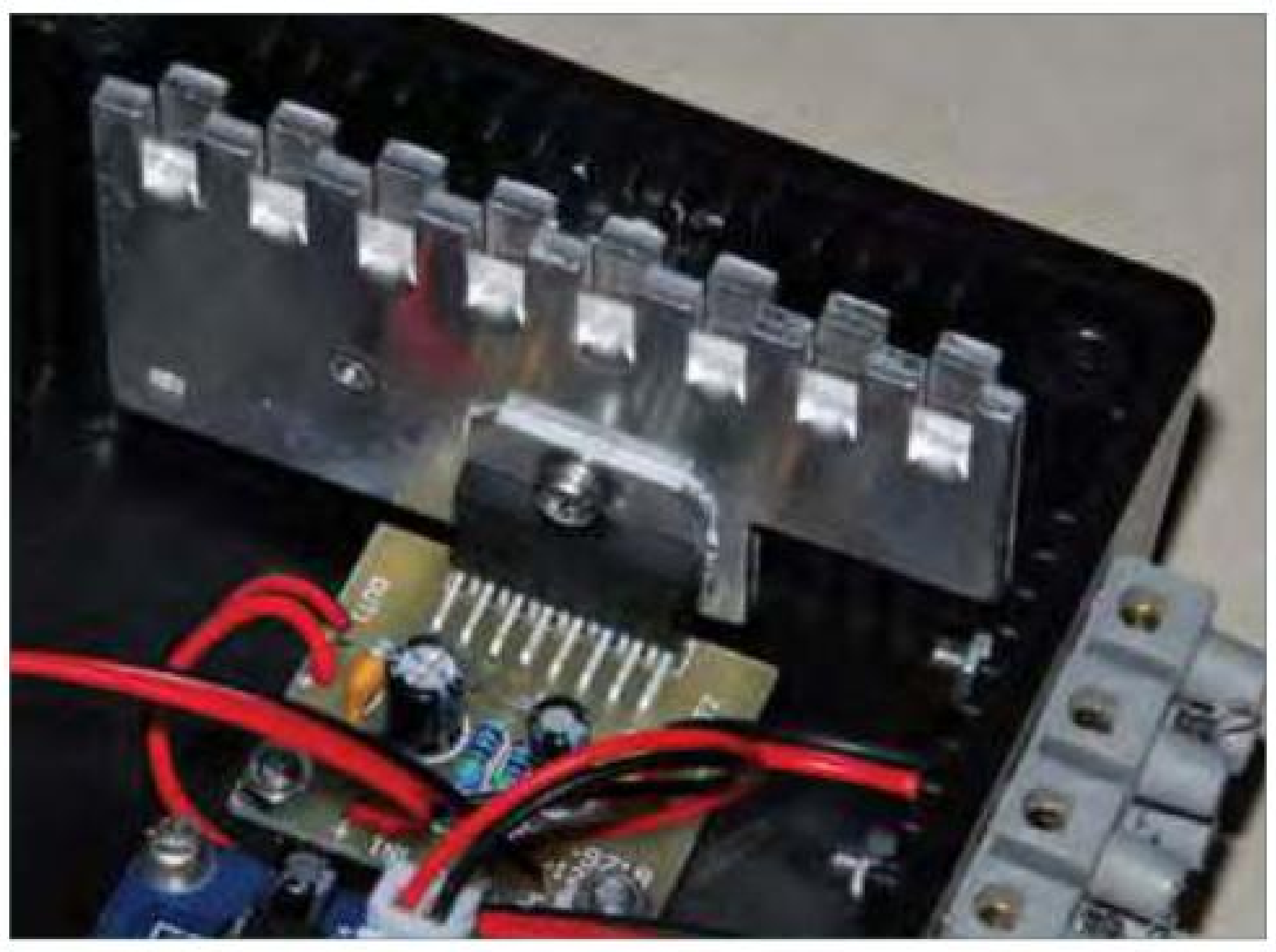


Fig.7. The TDA7297 amplifier requires plenty of heatsinking. Here, a small heatsink has been used, but it is fan-cooled. If a fan isn't used, the heatsink must be much larger.

Next, build (if required) the TDA7297 module and then make the following connections to the board:

- 12V (+) and (–)
- Speakers ×2 (marked as 'out')

The amplifier signal inputs are not quite as you might first think. In our application, the two speaker output wires from the ISD1820 voice module connect to the IN1 and IN2 connections of the amplifier board. The signal ground on the amplifier board is not connected. That is, the positive and negative voice module speaker outputs become the left and right inputs to the amplifier board. If you want plenty of volume (and why wouldn't you?), no volume control is needed on the amplifier input – the voice module output levels are a good match for the amplifier inputs.

Apply power to the system and your recorded message should boom through the connected speakers. Run the system for only a few moments in this form – without adequate cooling, the TDA7297 will get hot very quickly.

Final form

I built the components into a plastic box and fitted a relatively small heatsink to the amplifier module but used a small fan to draw air through the enclosure. If you choose not to use a fan, you will need a much larger heatsink than shown here. (Note that the TDA7297 chip is temperature protected and is rated to run at up to 70°C.)

Speaker and power connections were made via terminal strips attached to the outside of the box.

To run the system, you will need a plugpack power supply capable of 2A at about 12V. Of course, you could also use a battery supply (eg, a float-charged SLA battery), and this will allow the system to operate even if mains power is lost.

To trigger the warning, use a relay to feed power to the system when the right conditions occur. For example, smoke detectors are available with additional relay outputs, as are burglar alarms.

I used two horn speakers but you could use normal ceiling-mounted PA speakers, or even old Hi-Fi speakers. And there's also no need to use two speakers – in many applications, running just one will be sufficient.

Summary

In use, this is an extraordinarily effective alarm. Everyone who heard it in action – sirens blaring, ominous voice messages shouting – were utterly disconcerted. In fact, when I was testing the system, I needed to muffle the speakers so that neighbours didn't start running to my aid!

Practical Electronics PCB SERVICE

PROJECT	CODE	PRICE
JUNE 2018		
High Performance 10-Octave Stereo Graphic Equaliser	01105171	£15.30
JULY 2018		
Touchscreen Appliance Energy Meter – Part 1	04116061	£17.75
Automotive Sensor Modifier	05111161	£12.88
AUGUST 2018		
Universal Temperature Alarm	03105161	£7.05
Power Supply For Battery-Operated Valve Radios	18108171 18108172 18108173 18108174	£27.50
SEPTEMBER 2018		
3-Way Active Crossover	01108171	£22.60
Ultra-low-voltage Mini LED Flasher	16110161	£5.60
OCTOBER 2018		
6GHz+ Touchscreen Frequency Counter	04110171	£12.88
Two 230VAC MainsTimers	10108161 10108162	£12.88
NOVEMBER 2018		
Super-7 AM Radio Receiver	06111171	£27.50
FEBRUARY 2019		
1.5kW Induction Motor Speed Controller	10105122	£35.00
MARCH 2019		
10-LED Bargraph Main Board	04101181	£11.25
+Processing Board	04101182	£8.60
APRIL 2019		
Heater Controller	10104181	£14.00
MAY 2019		
2× 12V Battery Balancer	14106181	£5.60
Deluxe Frequency Switch	05104181	£10.45
USB Port Protector	07105181	£5.60
JUNE 2019		
Arduino-based LC Meter	04106181	£8.00
USB Flexitimer	19106181	£10.45
JULY 2019		
Full-wave 10A Universal Motor Speed Controller	10102181	£12.90
Recurring Event Reminder	19107181	£8.00
Temperature Switch Mk2	05105181	£10.45
AUGUST 2019		
Brainwave Monitor	25108181	£12.90
Super Digital Sound Effects Module	01107181	£5.60
Watchdog Alarm	03107181	£8.00
PE Theremin (three boards: pitch, volume, VCA)	PETX0819	£19.50
PE Theremin component pack (see p.56, August 2019)	PETY0819	£15.00
OCTOBER 2019		
Programmable GPS-synced Frequency Reference	04107181	£11.50
Digital Command Control Programmer for Decoders	09107181	£8.75
Opto-isolated Mains Relay (main board)	10107181	£11.50
Opto-isolated Mains Relay (2 × terminal extension board)	10107182	
NOVEMBER 2019		
Tinnitus & Insomnia Killer (Jaycar case – see text)	01110181	£8.75
Tinnitus & Insomnia Killer (Altronics case – see text)	01110182	£8.75

PROJECT	CODE	PRICE
DECEMBER 2019		
Extremely Sensitive Magnetometer	04101011	£16.75
Four-channel High-current DC Fan and Pump Controller	05108181	£8.75
Useless Box	08111181	£11.50
JANUARY 2020		
Isolated Serial Link	24107181	£8.50
FEBRUARY 2020		
Motion-Sensing 12V Power Switch	05102191	5.95
USB Keyboard / Mouse Adaptor	24311181	8.50
DSP Active Crossover (ADC)	01106191	45.00
DSP Active Crossover (DAC) ×2	01106192	
DSP Active Crossover (CPU)	01106193	
DSP Active Crossover (Power/routing)	01106194	
DSP Active Crossover (Front panel)	01106195	
DSP Active Crossover (LCD)	01106196	
MARCH 2020		
Diode Curve Plotter	04112181	£10.95
Steam Train Whistle / Diesel Horn Sound Generator	09106181	£8.50
Universal Passive Crossover (one off)	UPC0320	£12.50
Crossover component set for Wavecor speakers (one off)	WAVXO	(see website)

PE/EPE PCB SERVICE

Order Code	Project	Quantity	Price

Name ....

Address ....

....

Tel ....

Email ....

I enclose payment of £ .... (cheque/PO in **£ sterling only**)

payable to: **Practical Electronics**

Card No ....

Valid From ....Expiry Date ....

Card Security No ....

You can also order PCBs by phone, Fax, Email or via the shop on our website on a secure server: www.epemag.com

PCBs for most recent *PE*/*EPE* constructional projects are available. From the July 2013 issue onwards, PCBs with eight-digit codes have silk screen overlays and, where applicable, are double-sided, plated-through hole, with solder mask. They are similar to photos in the project articles. Earlier PCBs are likely to be more basic and may not include silk screen overlay, be single-sided, lack plated-through holes and solder mask.

Always check price and availability in the latest issue or online. A large number of older boards are listed for ordering on our website.

We do not supply kits or components for our projects. For older projects it is important to check the availability of all components before purchasing PCBs.

Back issues of articles are available – see Back Issues page for details.

All prices include VAT and UK p&p. Add £3 per *project* for post to Europe; £4 per *project* outside Europe. Orders and payment should be sent to:

Practical Electronics, Electron Publishing Ltd
113 Lynwood Drive, Merley, Wimborne, Dorset BH21 1UU
Tel 01202 880299 Email: shop@electronpublishing.com
On-line Shop: www.epemag.com

Cheques should be made payable to 'Practical Electronics' **(Payment in £ sterling only)**.

NOTE: While 95% of our boards are held in stock and are dispatched within seven days of receipt of order, please allow a maximum of 28 days for delivery if we need to restock.

CLASSIFIED ADVERTISING

Practical Electronics

If you want your advertisements to be seen by the largest readership at the most economical price then our classified page offers excellent value. The rate for semi-display space is £10 (+VAT) per centimetre high, with a minimum height of 2·5cm. All semi-display adverts have a width of 5.5cm. The prepaid rate for classified adverts is 40p (+VAT) per word (minimum 12 words).

Cheques are made payable to 'Practical Electronics'. **VAT must be added**. Advertisements with remittance should be sent to: Practical Electronics, 113 Lynwood Drive, Wimborne, Dorset, BH21 1UU.

Tel 01202 880299 Email: stewart.kearn@wimborne.co.uk

For rates and further information on display and classified advertising please contact our Advertisement Manager, Stewart Kearn – see below.

Practical Electronics reaches more UK readers than any other UK monthly hobby electronics magazine. Our sales figures prove it.

We have been the leading monthly magazine in this market for the last twenty-seven years.

BOWOOD ELECTRONICS LTD
Suppliers of Electronic Components
www.bowood-electronics.co.uk
Unit 10, Boythorpe Business Park, Dock Walk, Chesterfield, Derbyshire S40 2QR. Sales: 01246 200 222
Send large letter stamp for Catalogue

The British Amateur Electronic Club at:
baec.tripod.com
Has many interesting articles on computers; digital electronics and analogue electronics.

COAST ELECTRONICS
BREAKOUTS-COMPONENTS-CONTRACT DESIGN-3D PRINTER PARTS-MUSICAL-MICROCONTROLLERS
WWW.COASTELECTRONICS.CO.UK

MISCELLANEOUS

VALVES AND ALLIED COMPONENTS IN STOCK. Phone for free list. Valves, books and magazines wanted. Geoff Davies (Radio), tel. 01788 574774.

PIC DEVELOPMENT KITS, DTMF kits and modules, CTCSS Encoder and Decoder/Display kits.
Visit www.cstech.co.uk

Electrical Industries Charity (EIC)
We help people working in the electrical, electronics and energy community as well as their family members and retirees.
We use workplace programmes that give the industry access to financial grants and a comprehensive range of free and confidential services.
www.electricalcharity.org

Andrew Kenny – Qualified Patent Agent
EPO UKIPO USPTO
Circuits Electric Machinery Mechatronics
Web: www.akennypatentm.com
Email: Enquiries@akennypatentm.com
Tel: 0789 606 9725

ADVERTISING INDEX

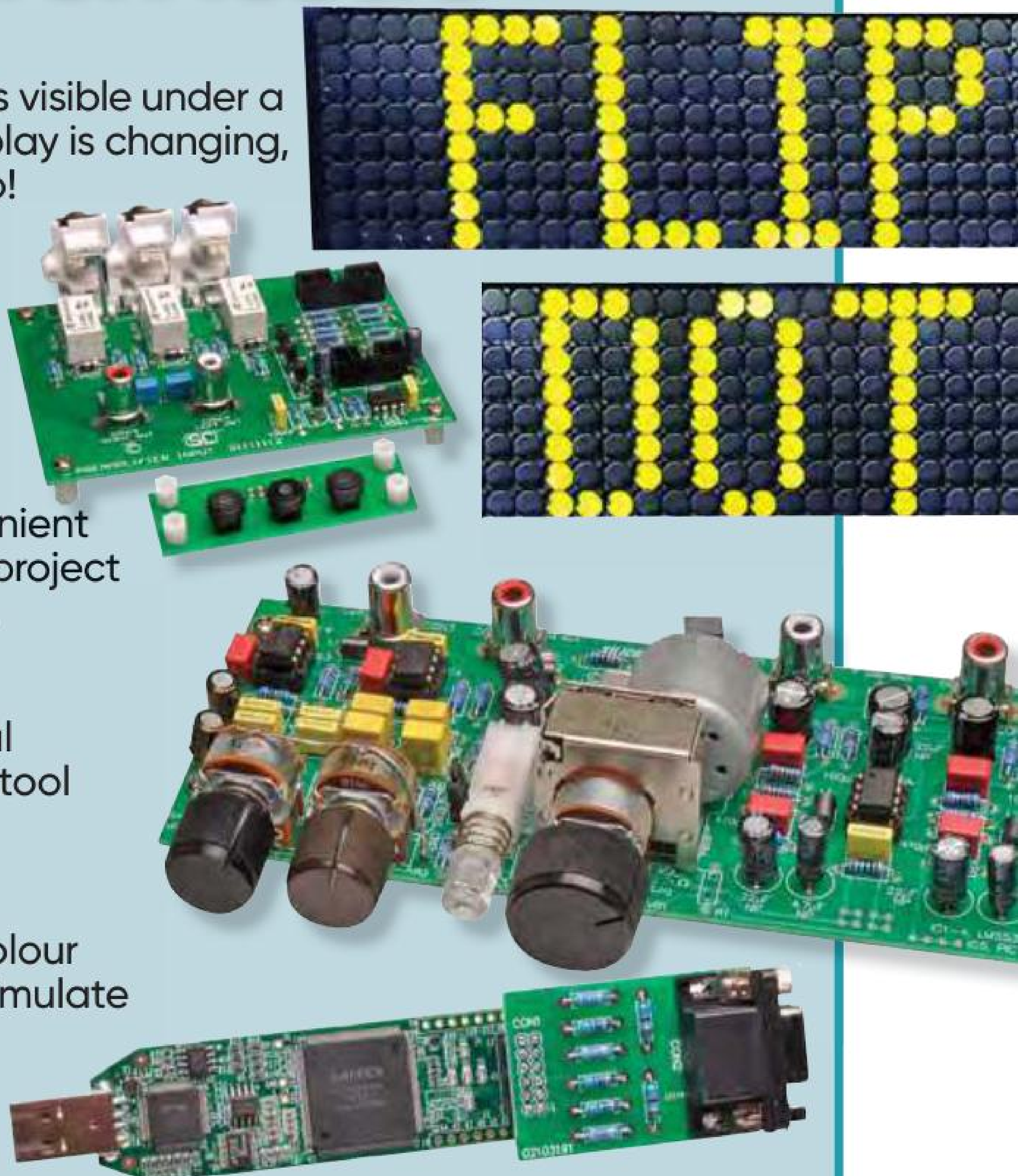
CRICKLEWOOD ELECTRONICS	52
ESR ELECTRONIC COMPONENTS	29
HAMMOND ELECTRONICS Ltd	8
IN SEARCH OF SIMULACRA	9
JPG ELECTRONICS	80
MICROCHIP	Cover (ii), Cover (iii), 5
PEAK ELECTRONIC DESIGN	Cover (iv)
POLABS D.O.O.	39
QUASAR ELECTRONICS	2
STEWART OF READING	9
TAG-CONNECT	29

Advertisement offices
Electron Publishing Ltd
113 Lynwood Drive
Merley
Wimborne,
Dorset BH21 1UU

Tel 01202 880299
Fax 01202 843233
Email stewart.kearn@wimborne.co.uk
Web www.epemag.com

For editorial contact details see page 7.

Next Month – in the March issue



Flip-dot display

If you want a dot matrix display which has digits/letters over 90mm high, is visible under a wide range of lighting conditions and uses no power except when the display is changing, then our new and very cool FLIP-DOT display is for you. It sounds great too!

Ultra-low-distortion Preamplifier with Tone Controls

This new design not only has superb controls but also has infrared remote volume control, input switching and muting. Meet the fantastic 2020 *Ultra-low-distortion Preamp!*

iCEstick VGA Terminal

Want to give a project the retro-computer look? Or do you need a convenient way to display a screen full of text on a low-cost monitor? Then this nifty project is for you! It does this using a low-cost iCEstick FPGA development board.

Visual programming with XOD

In next month's Arduino project built with XOD we will construct a digital high-current DC ammeter for 12-24V circuits. It's a great easy-to-build tool for your work bench with fast response peak-hold reading.

Make it with Micromite

Next month, we will show you just how easy it is to add an inexpensive colour touchscreen to your *Micromite Keyring Computer*. We will then use it to simulate the classic board game *Mastermind* – all remotely controlled, of course!

PLUS!

All your favourite regular columns from *Audio Out, Cool Beans and Circuit Surgery,* to *Electronic Building Blocks*, *PIC n' Mix* and *Net Work*.

On sale 5 March 2020

Content may be subject to change

WELCOME TO JPG ELECTRONICS

Selling Electronics in Chesterfield for 29 Years

Open Monday to Friday 9am to 5:30pm
And Saturday 9:30am to 5pm

- Aerials, Satellite Dishes & LCD Brackets
- Audio Adaptors, Connectors & Leads
- BT, Broadband, Network & USB Leads
- Computer Memory, Hard Drives & Parts
- DJ Equipment, Lighting & Supplies
- Extensive Electronic Components
  - *ICs, Project Boxes, Relays & Resistors*
- Raspberry Pi & Arduino Products
- Replacement Laptop Power Supplies
- Batteries, Fuses, Glue, Tools & Lots more...

T: 01246 211 202
E: sales@jpgelectronics.com
JPG Electronics, Shaw's Row,
Old Road, Chesterfield, S40 2RB
W: www.jpgelectronics.com

JPG Electronics

Retail & Trade Welcome • Free Parking • Google St View Tour: S40 2RB

NEW subscriptions hotline!

Practical Electronics

We have changed the way we sell and renew subscriptions. We now use 'Select Publisher Services' for all print subscriptions – to start a new subscription or renew an existing one you have three choices:

1. Call our NEW print subscription hotline: 01202 087631, or email: pesubs@selectps.com
2. Visit our online shop at: www.epemag.com
3. Send a cheque (payable to: 'Practical Electronics') with your details to:

   Practical Electronics Subscriptions, PO Box 6337,
   Bournemouth BH1 9EH, United Kingdom

Remember, we print the date of the last issue of your current subscription in a box on the address sheet that comes with your copy.

Digital subscribers, please continue to call 01202 880299 or visit: www.epemag.com

Published on approximately the first Thursday of each month by Electron Publishing Limited, 1 Buckingham Road, Brighton, East Sussex BN1 3RA. Printed in England by Acorn Web Offset Ltd., Normanton WF6 1TW. Distributed by Seymour, 86 Newman St., London W1T 3EX. Subscriptions UK: £26.99 (6 months); £49.85 (12 months); £94.99 (2 years). EUROPE: airmail service, £30.99 (6 months); £57.99 (12 months); £109.99 (2 years). REST OF THE WORLD: airmail service, £37.99 (6 months); £70.99 (12 months); £135.99 (2 years). Payments payable to 'Practical Electronics', Practical Electronics Subscriptions, PO Box 6337, Bournemouth BH1 9EH, United Kingdom. Email: pesubs@selectps.com. PRACTICAL ELECTRONICS is sold subject to the following conditions, namely that it shall not, without the written consent of the Publishers first having been given, be lent, resold, hired out or otherwise disposed of by way of Trade at more than the recommended selling price shown on the cover, and that it shall not be lent, resold, hired out or otherwise disposed of in a mutilated condition or in any unauthorised cover by way of Trade or affixed to or as part of any publication or advertising, literary or pictorial matter whatsoever.

Faster Programming, Wider Voltage Ranges and Enhanced Interface Options

PICkit™ 4 enables faster design development

MPLAB®
PICKIT 4
IN-CIRCUIT DEBUGGER
MICROCHIP

With five times faster programming and a wider 1.2V to 5V range, the Microchip MPLAB® PICkit™ 4 Development Tool supports low-cost development of even more Microchip controllers.

PICkit™ 4 also introduces improved USB connectivity with a hi-speed USB 2.0 interface and advanced debugging over 4-wire JTAG and serial wire debug with streaming data gateway in addition to legacy interfaces.

The new advanced interfaces support in-circuit programming and debugging of Microchip's CEC1702 hardware cryptography-enabled MCUs, in addition to dsPIC® Digital Signal Controllers and PIC® microcontrollers from 8- to 32-bit.

MICROCHIP

www.microchip.com/PICkit4eu

The Microchip name and logo, the Microchip logo, MPLAB, PIC and dsPIC are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries. PICkit and In-Circuit Serial Programming (ISCP) are trademarks of Microchip Technology Inc. in the U.S.A. and other countries. All other trademarks mentioned herein are the property of their respective companies. © 2018 Microchip Technology Inc. All rights reserved.
DS-50002745A. MEC2206Eng04/18

PEAK®
electronic design ltd

Celebrating the return of Practical Electronics!
Limited time offer only

Use code PE10 during checkout
(or on the phone!)

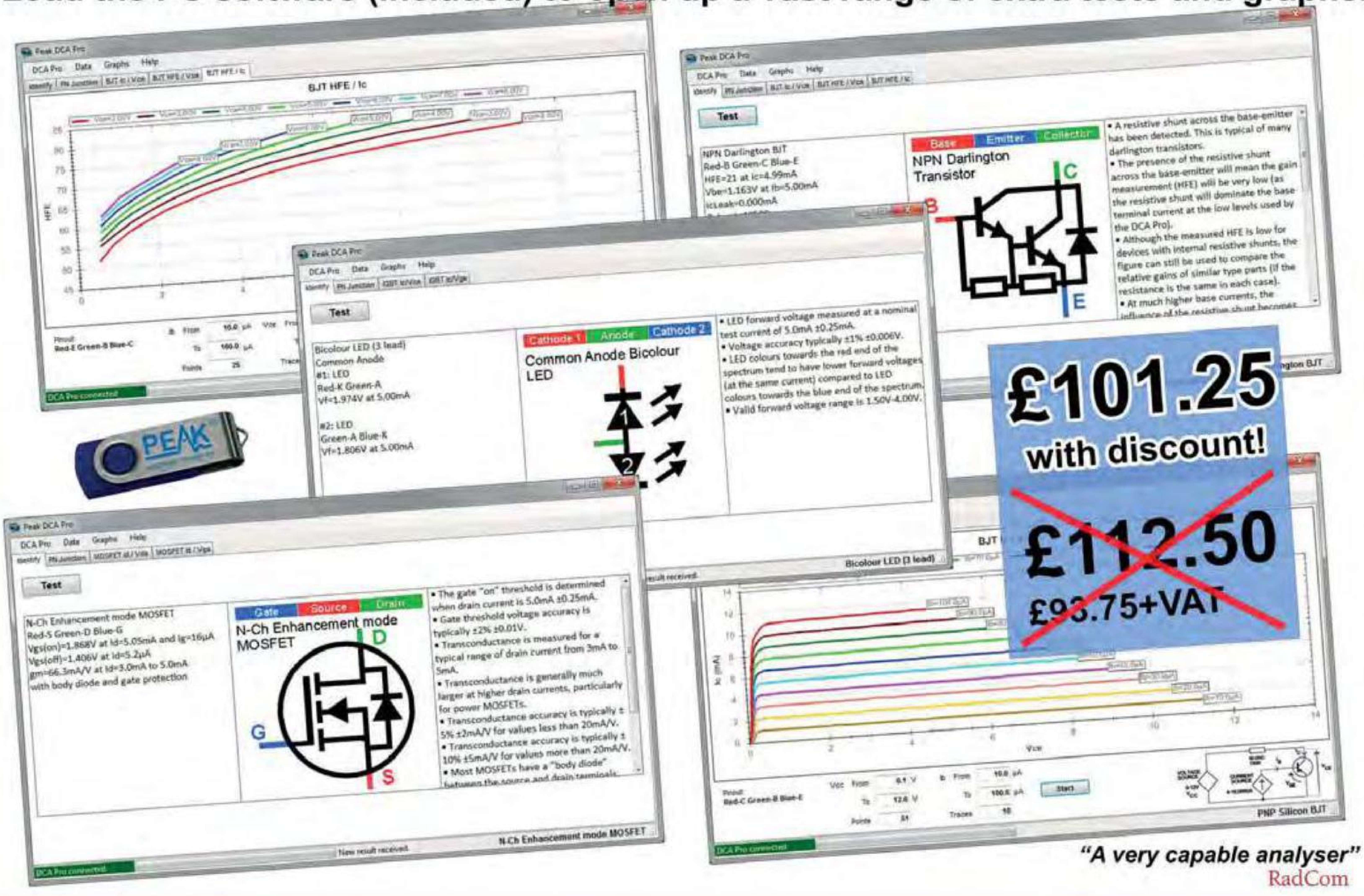
DCA75

Advanced Semiconductor Analyser and Curve Tracer

atlas DCA Pro
DCA75
USB + Graphics
PEAK®
electronic design ltd
on - test
scroll - off
warning: avoid charge/voltages

A versatile instrument that you'll love using for testing a huge range of semiconductor components.
Supports MOSFETs, transistors, diodes, LEDs, JFETs, IGBTs and more.
Connect any way round and let the ***DCA Pro*** identify the component type, pinout and lots of parameters.
Can be used standalone or connected to a Windows® PC (XP up to Win10).

Load the PC software (included) to open up a vast range of extra tests and graphs.


Peak DCA Pro
DCA Pro
Data
Graphs
Help
BJT HFE / Ic
HFE
Test
Base
Emitter
Collector
NPN Darlington Transistor
B
C
E
Cathode 1
Anode
Cathode 2
Common Anode Bicolour LED
1
2
Gate
Source
Drain
N-Ch Enhancement mode MOSFET
G
D
S
Ic (mA)
Vce
PNP Silicon BJT
DCA Pro connected
New result received
PEAK

NPN Darlington BJT
Red-B Green-C Blue-E
HFE=21 at Ic=4.99mA
Vbe=1.163V at Ib=5.00mA
IcLeak=0.000mA

- A resistive shunt across the base-emitter has been detected. This is typical of many darlington transistors.
- The presence of the resistive shunt across the base-emitter will mean the gain measurement (HFE) will be very low (as the resistive shunt will dominate the base terminal current at the low levels used by the DCA Pro).
- Although the measured HFE is low for devices with internal resistive shunts, the figure can still be used to compare the relative gains of similar type parts (if the resistance is the same in each case).
- At much higher base currents, the

Bicolour LED (3 lead)
Common Anode
#1: LED
Red-K Green-A
Vf=1.974V at 5.00mA

#2: LED
Green-A Blue-K
Vf=1.806V at 5.00mA

- LED forward voltage measured at a nominal test current of 5.0mA ±0.25mA.
- Voltage accuracy typically ±1% ±0.006V.
- LED colours towards the red end of the spectrum tend to have lower forward voltages (at the same current) compared to LED colours towards the blue end of the spectrum.
- Valid forward voltage range is 1.50V-4.00V.

N-Ch Enhancement mode MOSFET
Red-S Green-D Blue-G
Vgs(on)=1.868V at Id=5.05mA and Ig=16µA
Vgs(off)=1.406V at Id=5.2µA
gm=66.3mA/V at Id=3.0mA to 5.0mA
with body diode and gate protection

- The gate "on" threshold is determined when drain current is 5.0mA ±0.25mA.
- Gate threshold voltage accuracy is typically ±2% ±0.01V.
- Transconductance is measured for a typical range of drain current from 3mA to 5mA.
- Transconductance is generally much larger at higher drain currents, particularly for power MOSFETs.
- Transconductance accuracy is typically ± 5% ±2mA/V for values less than 20mA/V.
- Transconductance accuracy is typically ± 10% ±5mA/V for values more than 20mA/V.
- Most MOSFETs have a "body diode"

£101.25
with discount!
~~**£112.50**~~
~~£93.75+VAT~~

"A very capable analyser"
RadCom

It's only possible to show summary specifications here. Please ask if you'd like detailed data. Further information is also available on our website. Product price refunded if you're not happy.

Tel. 01298 70012
www.peakelec.co.uk
sales@peakelec.co.uk

Atlas House, 2 Kiln Lane
Harpur Hill Business Park
Buxton, Derbyshire
SK17 9JL, UK

UK designed **and** manufactured